2006-09

Design and integration of a three degrees-of freedom robotic vehicle with control moment gyro for the Autonomous Multiagent Physically Interacting Spacecraft (AMPHIS) testbed

Hall, Jason S.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

DESIGN AND INTEGRATION OF A THREE DEGREES-OF-FREEDOM ROBOTIC VEHICLE WITH CONTROL MOMENT GYRO FOR THE AUTONOMOUS MULTI-AGENT PHYSICALLY INTERACTING SPACECRAFT (AMPHIS) TESTBED

by

Jason S. Hall

September 2006

Thesis Advisor: Marcello Romano

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE		Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 2006	Master's Thesis

4. TITLE AND SUBTITLE Design and Integration of a Three Degrees-of-Freedom Robotic Vehicle with Control Moment Gyro for the Autonomous Multi-agent Physically Interacting Spacecraft (AMPHIS) Testbed	5. FUNDING NUMBERS
6. AUTHOR Jason S. Hall	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Approved for public release; distribution is unlimited	A

13. ABSTRACT

The use of fractionated spacecraft systems in on-orbit spacecraft assembly has the potential to provide benefits to both the defense and civil space community. To this end, much research must be conducted to develop and prove the requisite technologies to achieve these benefits. This thesis contributes to that effort by presenting the design and system integration, operating procedures and software development for a prototype three Degrees-Of-Freedom (DOF) Spacecraft Simulator. This simulator will be used in the Proximity Operations Simulator Facility, as part of the Naval Postgraduate School's Spacecraft Robotics Laboratory, to simulate autonomous guidance, navigation and control (GNC) for spacecraft proximity operations and assembly within the framework of the Autonomous Multi-Agent Physically Interacting Spacecraft project. The new spacecraft simulator includes several key enhancements over the previous Autonomous Docking and Spacecraft Servicing Simulator (AUDASS) developed in 2005 including a smaller and more agile structure, reduced air consumption and a Miniature Single-Gimbaled Control-Moment-Gyroscope (MSGCMG) to provide the necessary torque about the rotation axis.

The MSGCMG in the simulator is a low-cost, low-mass, easily controlled momentum exchange device with a high torque to required power ratio. Furthermore, it provides the vehicle with high slew-rate capability, a key measure of performance in on-orbit spacecraft assembly. Simulation and experimental results are presented for the prototype AMPHIS vehicle, demonstrating a potential slew-rate of 4.8 deg/s for a 30 degree rest-to-rest maneuver.

The ultimate goal of this thesis is to provide the design specifications, combined with the necessary documentation and software development, for the prototype vehicle of the testbed for the AMPHIS project. The work conducted in fabricating the prototype vehicle will enable rapid fabrication of two additional vehicles which will provide an essential hardware-in-the-loop capability for experimentation with evolving control algorithms, sensors and mating mechanisms to be used for autonomous spacecraft assembly.

14. SUBJECT TERMS Autonomous on-orbit spacecraft assembly, fractionated spacecraft, miniature control-moment-gyro, proximity operations, multi-agent, robotic, hardware-in-the-loop testbed	15. NUMBER OF PAGES 93
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

DESIGN AND INTEGRATION OF A THREE DEGREES-OF-FREEDOM ROBOTIC VEHICLE WITH CONTROL MOMENT GYRO FOR THE AUTONOMOUS MULTI-AGENT PHYSICALLY INTERACTING SPACECRAFT (AMPHIS) TESTBED

Jason S. Hall
Lieutenant Commander, United States Navy
B.S., United States Naval Academy, 1997

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ASTRONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 2006

Author: Jason S. Hall

Approved by: Marcello Romano
Thesis Advisor

Anthony J. Healey
Chairman, Department of Mechanical and Astronautical Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The use of fractionated spacecraft systems in on-orbit spacecraft assembly has the potential to provide benefits to both the defense and civil space community. To this end, much research must be conducted to develop and prove the requisite technologies to achieve these benefits. This thesis contributes to that effort by presenting the design and system integration, operating procedures and software development for a prototype three Degrees-Of-Freedom (DOF) Spacecraft Simulator. This simulator will be used in the Proximity Operations Simulator Facility, as part of the Naval Postgraduate School's Spacecraft Robotics Laboratory, to simulate autonomous guidance, navigation and control (GNC) for spacecraft proximity operations and assembly within the framework of the Autonomous Multi-Agent Physically Interacting Spacecraft project. The new spacecraft simulator includes several key enhancements over the previous Autonomous Docking and Spacecraft Servicing Simulator (AUDASS) developed in 2005 including a smaller and more agile structure, reduced air consumption and a Miniature Single-Gimbaled Control-Moment-Gyroscope (MSGCMG) to provide the necessary torque about the rotation axis.

The MSGCMG in the simulator is a low-cost, low-mass, easily controlled momentum exchange device with a high torque to required power ratio. Furthermore, it provides the vehicle with high slew-rate capability, a key measure of performance in on-orbit spacecraft assembly. Simulation and experimental results are presented for the prototype AMPHIS vehicle, demonstrating a potential slew-rate of 4.8 deg/s for a 30 degree rest-to-rest maneuver.

The ultimate goal of this thesis is to provide the design specifications, combined with the necessary documentation and software development, for the prototype vehicle of the testbed for the AMPHIS project. The work conducted in fabricating the prototype vehicle will enable rapid fabrication of two additional vehicles which will provide an essential hardware-in-the-loop capability for experimentation with evolving control algorithms, sensors and mating mechanisms to be used for autonomous spacecraft assembly.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS

AMPHIS	-	Autonomous Multi-agent Physically Interacting Spacecraft
NPS	-	Naval Postgraduate School
SSAG	-	Space Systems Academic Group
SRL	-	Spacecraft Robotics Laboratory
POSF	-	Proximity Operations Simulator Facility
DOF	-	Degrees of Freedom
CMG	-	Control Moment Gyro
MSGCMG	-	Miniature Single Gimbaled Control Moment Gyro
iGPS	-	Indoor Global Positioning System
GNC	-	Guidance, Navigation and Control
LiDAR	-	Light Detection and Ranging
QD	-	Quick Disconnect
OD	-	Outer Diameter
ID	-	Inner Diameter
NPT	-	National Pipe tapered Thread
RS-232	-	Recommended Standard -232
TCP/IP	-	Transmission Control Protocol/Internet Protocol
CPU	-	Central Processing Unit
LEO	-	Low Earth Orbit
I/O	-	Input/Output
ISA	-	Industry Standard Architecture
PCI	-	Peripheral Component Interface
USB	-	Universal Serial Bus
LPT	-	Line Printing Terminal
KVM	-	Keyboard, Video, Mouse
DRAM	-	Dynamic Random Access Memory
CMOS	-	Complementary Metal Oxide Semiconductor
SCBA	-	Self-Contained Breathing Apparatus

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

The author would like to acknowledge the financial support of the Naval Postgraduate School.

The author would also like to thank the following for their invaluable assistance in the completion of this thesis:

My beautiful wife Meredith and two wonderful children Carter and Morgen, who have kept me smiling despite the long hours away from home.

Mr. Bob Reehm who has continued to remind me the purpose of it all, keeping me spiritually grounded.

Dr. Marcello Romano for his expertise and guidance throughout the thesis process.

LCDR Blake Eikenberry and LT Bill Price for their friendship and partnership thus far in the developmental process of the AMPHIS testbed and for what they still will provide.

Captain Dave Friedman and LCDR Tracy Shay for their discoveries of many best-practices in component selection and robotic vehicle construction.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. POTENTIAL BENEFIT OF FRACTIONATED SPACECRAFT SYSTEMS

Since the beginning of the space age, the U.S. has invested trillions of dollars to build, launch and operate its space systems. Although these systems do provide unquestioned benefits to both the general public and the U.S. Government, there have been increased efforts recently to reduce both their acquisition and lifetime costs. Dr. Pedro "Pete" L. Rustan, the Director of Advanced Systems and Technology at the National Reconnaissance Office, speaking to the U.S. House Armed Services Committee in July 2005, pointed to ten major problems of the current space systems acquisition process. These problems are:

1. Overly detailed requirements with little flexibility
2. Proceeding to acquisition before proper technology maturity
3. Insufficient budget flexibility
4. Requirements creep
5. Management experience shortfalls resulting in excess personnel in program offices
6. Incomplete requirements flow down to subsystems to meet expected performance
7. Failure to properly manage subcontractors
8. Uncertainties in the expected performance and schedule of the new generation of electronic components
9. Tendency to build a new spacecraft for each new set of requirements
10. Responsibility for the development of only on-orbit capabilities exclusive of requisite ground services. (Ref. [1])

One potential solution to a majority of these problems involves a paradigm shift in spacecraft architecture development from the traditional one-time use, single purpose, tailored designed systems to one involving fractionated or reconfigurable systems. The

rapidly evolving field of fractionated spacecraft research involves both heterogeneous and homogeneous systems. Homogeneous fractionated spacecraft systems are composed of several identical interacting agents that are small-scale replicas of a traditional large spacecraft and are fully capable of functioning independently of each other. Heterogeneous systems, on the other hand, are composed of multiple interacting agents, each having a particular contribution to the group similar to biological organs. Although both types of fractionated spacecraft have their benefits and distracters, heterogeneous fractionated spacecraft, with their functional segmentation, have the unique ability to take advantage of more technological advances. (Ref. [2]) In principle, by selecting heterogeneous fractionated spacecraft over traditional large spacecraft architectures, many of the space acquisition failures highlighted by Dr. Rustan might be remedied. A detailed assessment of heterogeneous fractionated spacecraft, conducted by C. Mathieu and A.L. Weigel at the Massachusetts Institute of Technology (MIT) in the areas of maintainability, scalability, flexibility and responsiveness, substantiate this postulate (Ref. [2])

The removal of significant intra-modular reliance in fractionated systems provides for a parallel design, fabrication, assembly and validation process. (Ref. [2]) This flexibility provides a solution to many of the Dr. Rustan's points. First, the heterogeneous nature allows for successful program completion even in the presence of detailed requirements or a high degree of requirements creep. Second, it provides a means of early detection and correction of improperly designed subsystems with respect to the expected system performance. Third, it specifically allows for the building of a new spacecraft for each new requirement due to the heterogeneous nature of its modular design.

Typically, the smaller nature of fractionated systems allows for each module to be placed into orbit separately on more affordable launch platforms. (Ref. [2]) This capability further improves the space acquisition process for a couple key reasons. First, it can alleviate much of the budget inflexibility due to its significant reduction of the launch cost which is traditionally the most costly line item in any space program's budget. Second, it allows for less reliance on the ground services and replaces the focus on the development of on-orbit capabilities.

B. CURRENT FRACTIONATED SPACECRAFT RESEARCH FOR ON-ORBIT ASSEMBLY

With such a potential for enormous benefits from using fractionated systems comes the need to prove the requisite technologies to achieve this benefits. There currently exist several research and academic institutions involved in pursuing this goal with both hardware-in-the-loop experimentation and computer modeling and simulation.

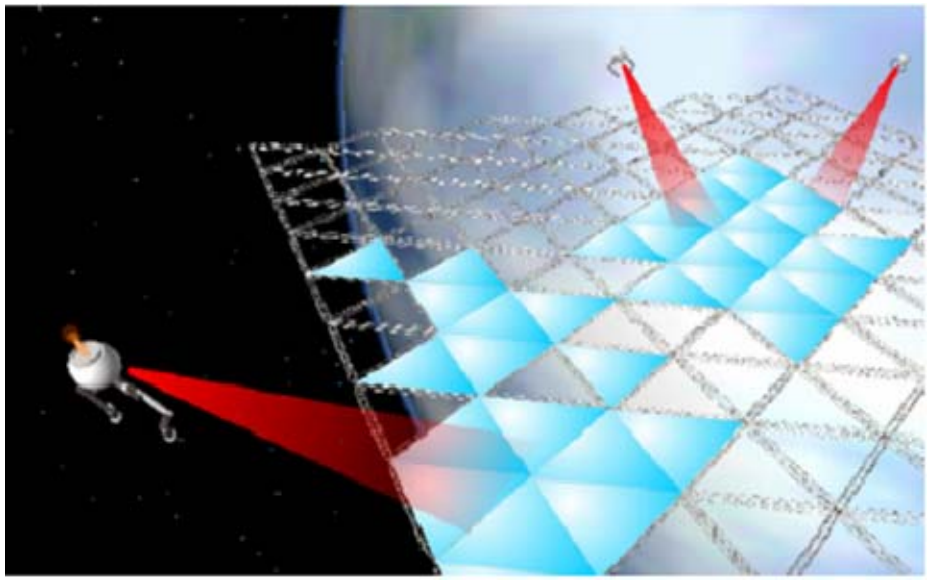

Figure 1. MIT Envisioned Autonomous Assembly Using Raster-Scanning Range Imagers (Ref. [4])

MIT's Field and Space Robotics Laboratory (FSRL) is currently researching the planning and control of space systems in collaboration with the Japanese Aerospace Exploration Agency (JAXA). This area of research is focused on development of sensing strategies for determination of both known and unknown targets, planning algorithms for safe trajectory and manipulation schemes, control algorithms to affect these trajectories and manipulation schemes for multiple robot cooperative approaches. (Ref. [3]) This research has lead to the development of a method for estimating the requisite parameters for a given dynamic space structure by using asynchronous raster-scanning range imagers given an a priori knowledge of the mode shapes of the structure. (Ref. [4]) Figure 1 illustrates the potential application of this method for multiple robotic vehicles in relation

to an on-orbit structure. In addition to its collaboration with MIT, JAXA has developed and demonstrated robotic assembly for massive on-orbit structures such as the Space Solar Power System (SSPS). The SSPS robot testbed was fabricated to validate the assembly and maintenance by autonomous multiple robot systems. Figure 2 shows the ground testbed used for multiple experiments related to flexible structure capture, locomotion on lightweight structures, inflatable structure assembly, deployable structure assist, transportation and component insertion and large-scale structure assembly. (Ref. [5]) These experiments demonstrate a potential inability for one small robotic vehicle to be able to follow the motion of a large flexible structure. Additionally, they reveal an inability for a single large robotic vehicle to connect to the structure given the exerted docking forces. It further postulates that a cooperative robotic vehicle system might solve this problem. (Ref. [6])

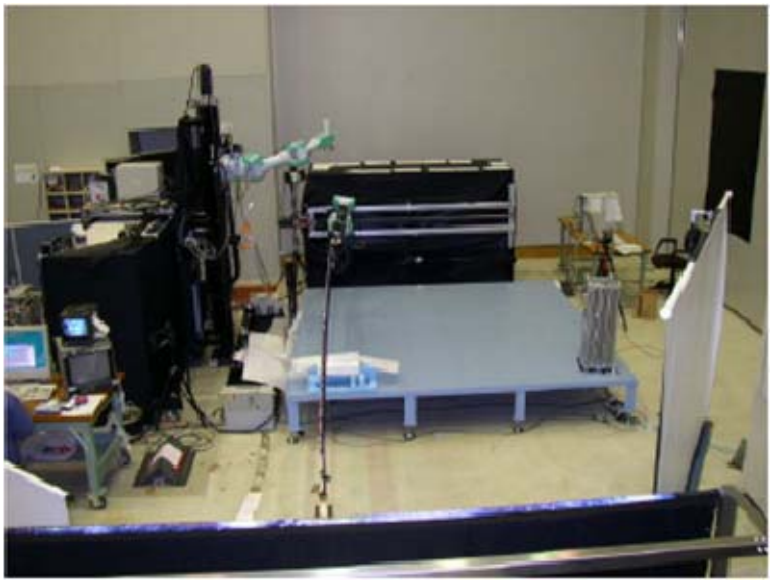

Figure 2. JAXA Ground Testbed for SSPS. (Refs. [5],[6])

Figure 3 illustrates the 6 Degrees of Freedom (DOF) capable Formation Control Testbed (FCT) constructed by the Jet Propulsion Laboratory to support technology demonstration and development of the requisite control algorithms for fractionated spacecraft interaction. Each FCT robotic vehicle consists of a star tracker, a CG balanced attitude platform containing three reaction wheels, 16 1N air thrusters, wireless Ethernet

and a compactPCI Bus PowerPC750 flight computer. A linear air bearing, three air pads and a vertical stage provide 6 DOF for each vehicle by 16 3000 PSI air tanks. Each of three robots stands 64.5 inches high with a diameter of 59.5 inches, weighs approximately 358 kg and operates in a 40 foot diameter room. (Ref. [7])

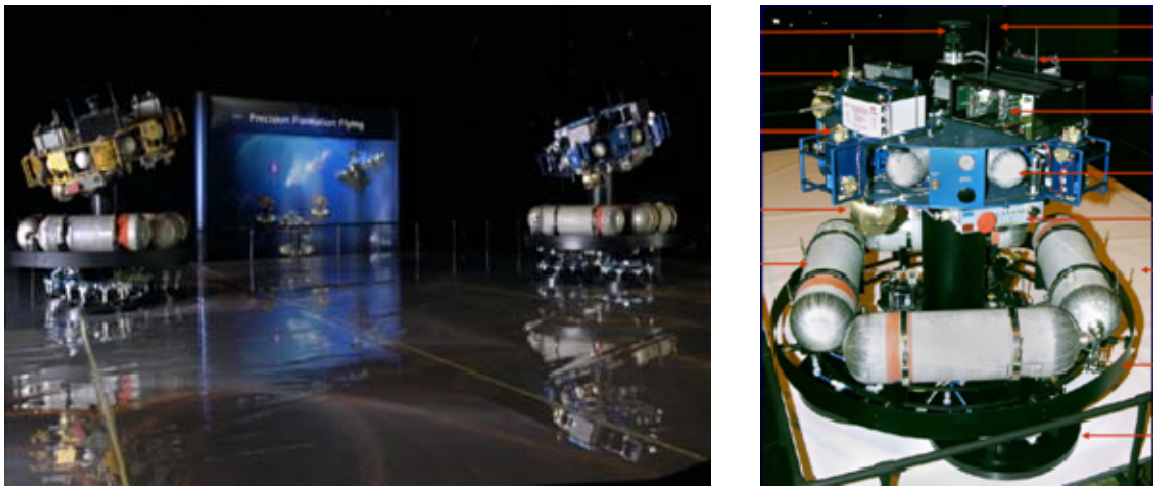

Figure 3. JPL Formation Control Testbed (Ref. [7])

C. MINIATURE CONTROL MOMENT GYROSCOPE BENEFITS AND CURRENT APPLICATIONS

Control Moment Gyros (CMGs) are well known as excellent momentum exchange devices, providing attitude control system solutions for missions with large momentum storage requirements and torque disturbances such as the International Space Station (ISS). However, despite the high torque to power ratio of CMGs, several practical and theoretical issues such as a complex control steering logic requirement in order to avoid singularities and reliability issues inherent to complex mechanical systems have prevented adoption as the primary attitude control actuator for most spacecraft. Recent advances in both the area of reliability and control have made a profound impact on these issues and wide-scale selection of CMG's as the primary momentum exchange device for both traditional and fractionated spacecraft is a near reality. (Refs. [8],[9],[10])

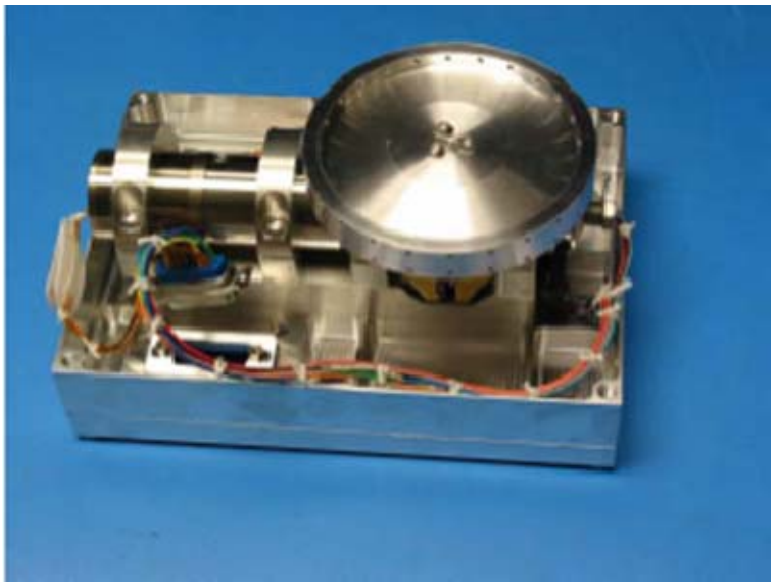

Figure 4. SSTL Miniature CMG in BILSAT-I (Ref. [11])

Figure 4 shows one of the two miniature Single Gimbaled CMGs (SGCMGs) currently on orbit in BILSAT I. This satellite, built by Surrey Satellite Technology Ltd. (SSTL) for Turkey, marks a first for carrying the smallest, commercially constructed CMGs to be flown on a spacecraft. (Refs. [11], [12]) Small robotic spacecraft have inherent power system restrictions due to their size, need to maneuver frequently and lack of deployable solar arrays. In comparison to other momentum exchange devices such as reaction wheels or momentum wheels, a CMG is much more power efficient. The power required for a perfect reaction wheel (assuming entirely mechanical, lossless system) is the product of its torque and the rotor spin speed which apply shaft power. Applying this to a typical reaction wheel with a rotor spin speed of 2500 rpm and a maximum torque of 162.4 mN-m, the power required is 42.5 W.

For CMGs, the input torque is virtually orthogonal to the rotor spin axis and therefore the shaft power is nearly zero if both the gimbal inertia and motor losses are assumed to be negligible. These assumptions are valid, particularly for a miniature SGCMG due to the individual component sizes. Given the same rotor from above and taking its rated momentum capacity of 20.3 Nms, and assuming a maximum gimbal acceleration of 1 rad/s^2, the peak power of the CMG is 620e-6 W.

The significant advantage for power conscious systems is quickly evident from the torque to power comparison between these two momentum exchange devices. It is for this primary reason that an investigation began into the merits of employing a SGCMG on the 3 DOF planar testbed in the Naval Postgraduate School's (NPS) Spacecraft Robotics Laboratory (SRL). After initial experimentation, a satisfactory design is presented.

D. RESEARCH CONDUCTED AT THE NAVAL POSTGRADUATE SCHOOL'S SPACECRAFT ROBOTICS LABORATORY

The SRL under the Department of Mechanical and Astronautical Engineering and the Space Systems Academic Group (SSAG) at NPS is also poised to study Guidance, Navigation and Control (GNC) approaches for fractionated spacecraft systems with both computer modeling and simulation and on-the-ground experimental validation on its Proximity Operations Simulator Facility (POSF). This facility, fully developed in 2005, comprises a 4.9 m by 4.3 m wide Epoxy Floor Surface with an average residual slope angle of approximately $2.6x10^{-3}$ deg. This residual slope translates to an average residual gravity acceleration of $1.4x10^{-3}$ m/s^2. (Ref. [13])

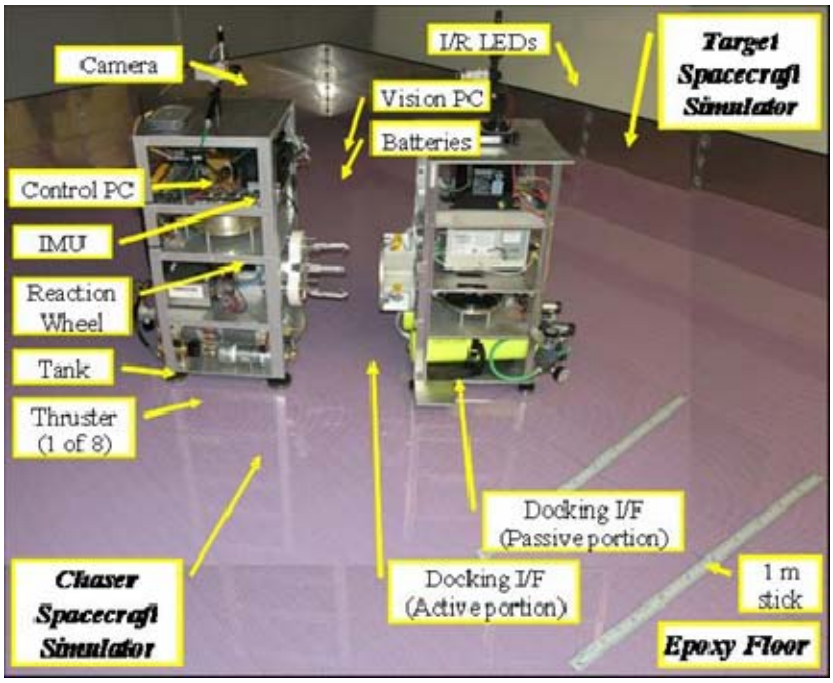

Figure 5. Autonomous Docking Testbed at the NPS SRL (Ref. [13])

Figure 5 shows the NPS Planar Autonomous Docking and Servicing Spacecraft Simulator (AUDASS). This system, built during 2004 and 2005, demonstrated a successful implementation of a vision based guidance and control systems to affect a rendezvous and docking scenario. (Refs. [13],[14],[15],[16]) The project was sponsored by the Air Force Research Laboratory (AFRL).

Research into the use of fractionated spacecraft systems with multiple physically interacting vehicles for on-orbit assembly continues today with the Autonomous Multi-agent Physically Interacting Spacecraft (AMPHIS) testbed project. This project is specifically focused on the study and experimental verification through hardware-in-the-loop testing of the necessary GNC approaches for these types of systems.

E. SCOPE OF THESIS

This thesis comprises the work involved to design and implement a proto-type spacecraft simulator with two translational DOFs and one attitude DOF for the AMPHIS testbed at the NPS SRL. It builds upon previous work performed in developing the AUDASS robotic vehicles with a particular focus on reducing the mass of the vehicles, increasing their endurance, allowing for different types of sensor mounting (i.e. Light Detection And Ranging (LiDAR), stereo vision or panoramic vision), and testing different actuators for both attitude and translational control. This work is an intermediate phase in the current effort towards being able to test computer-modeled reconfigurable spacecraft strategies in a simulated environment.

Specifically, this thesis presents four key improvements:

1. Light-weight, prefabricated aluminum structural members coupled with rigid plastic decking reduce the mass and enable rapid construction of the robotic vehicles to be used on the AMPHIS testbed.

2. A streamlined on-board air supply system increases available experimentation time.

3. A Miniature SGCMG (MSGCMG) provides a unique, low cost and high torque to mass and power ratio approach to rotational control for the AMPHIS simulators.

4. The refurbishment of a collocated air compressor significantly improves the existing POSF by providing rapid, low-cost refilling of the on-board air supply systems on both the AUDASS and AMPHIS simulators.

THIS PAGE INTENTIONALLY LEFT BLANK

II. THREE DEGREES-OF-FREEDOM (DOF) PROTOTYPE SPACECRAFT SIMULATOR DESIGN

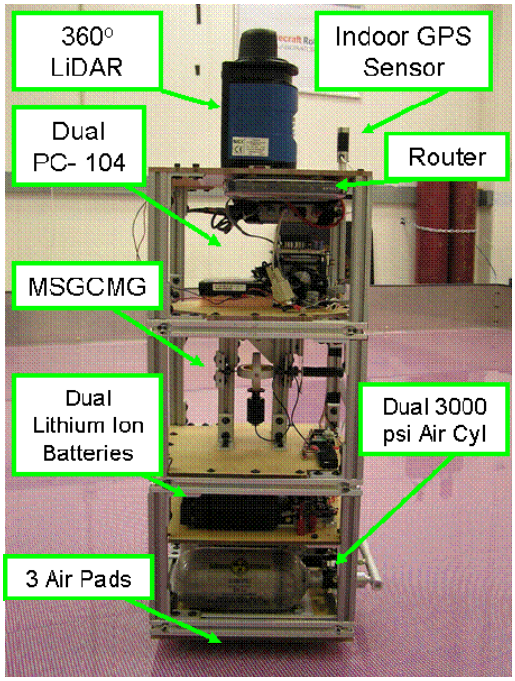

Figure 6. Prototype Spacecraft Simulator for the AMPHIS Testbed in the POSF

A. OVERVIEW OF AMPHIS TESTBED

The current AMPHIS testbed consists of a prototype Spacecraft Simulator as shown in Figure 6, a software development computer, and the components of the Proximity Operations Simulator Facility as described in the introduction to include two indoor GPS transmitters. Figure 7 illustrates this setup in schematic form to give the reader a sense for the hardware involved and the communication schemes employed between components. Each component of the three DOF AMPHIS prototype spacecraft simulator will be described in detail in the proceeding chapters.

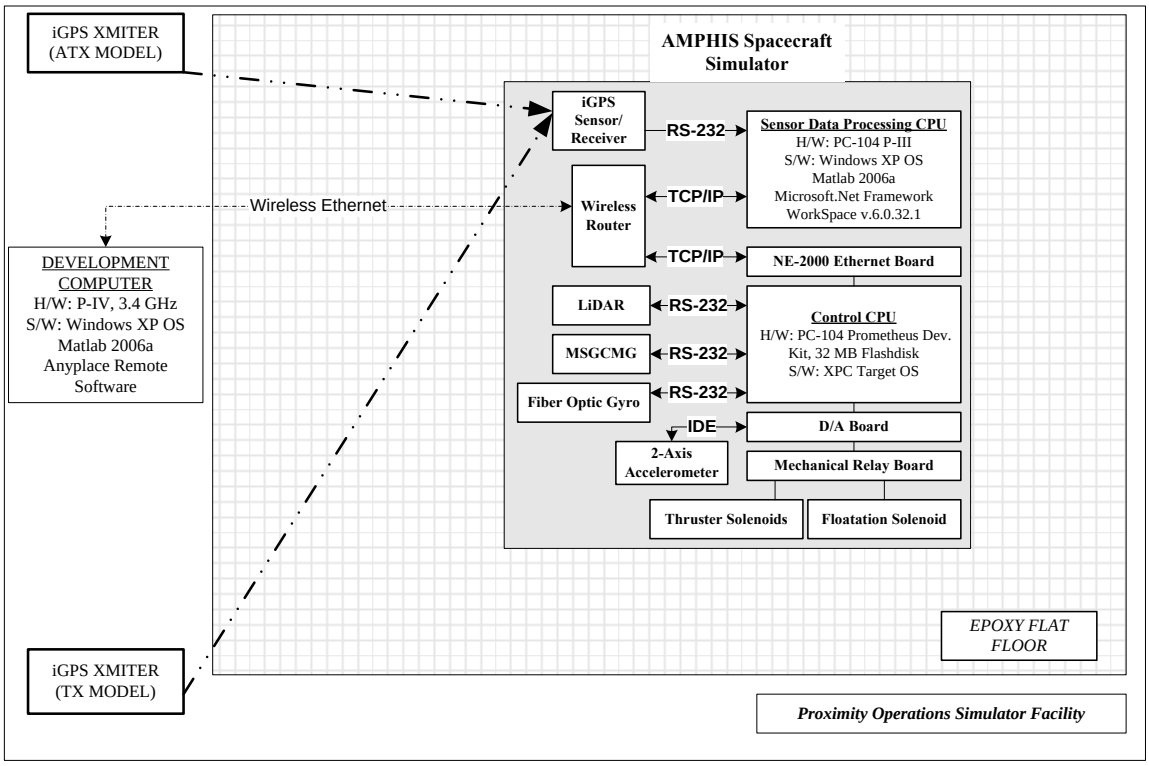

Figure 7. AMPHIS Testbed Schematic

B. FUNDAMENTAL DESIGN CONSIDERATIONS

In order to fully develop the concepts for multiple spacecraft assembly through hardware-in-the-loop testing, it is necessary to have sufficient space for maneuvering of the testbed vehicles. Given the constraints of the aforementioned dimensions for the flat floor in the POSF, the desire to have at least three robotic vehicles operating at once and the desire to test multiple sensor configurations drove the requirement for a significantly lighter and smaller vehicle. To this end, three key enhancements were achieved on the prototype spacecraft simulator for the AMPHIS testbed. These enhancements lead directly to a significant reduction in size and mass without sacrificing mobility or endurance while providing for rapid reconfiguration. Table 1 lists the key parameters on the prototype spacecraft simulator for the AMPHIS testbed. Manufacturer and limitation information for the components used on the simulator are contained within Appendix A.

Size	Length and Width	.30 [m]
	Height	.69 [m]
	Mass	23.95 [kg]
	Moment of Inertia about Z_{ch}	.75 [kg m^2]
Propulsion	Propellant	Air
	Equivalent Storage Capacity @ 21 [MPa] (3000 PSI)	.002 [m^3]
	Operating Pressure, Floating	0.35 [Mpa] (50 PSI)
	Operating Pressure, Thrusting	TBD
	Cont. Operation (No Thrust Factored)	75 [min]
	Thrust of Each Thruster	TBD
	CMG Max Torque	.334 [Nm]
	CMG Max Ang. Momentum	.049 [Nms]
Electrical & Electronic Subsystem	Battery Type	Lithium-Ion
	Storage Capacity	12 [Ah] @ 28[V]
	Computers	1 PC104 Pentium III 1 Prometheus
Sensors	Fiber Optic Gyro Bias	±20°/hr
	LiDAR Sensor	Under Development
	Optical Position Validation Sensor	Under Development
	Pseudo-GPS Sensor Accuracy	< .050 [mm]
	Accelerometers Bias Stability	± $8.5x10^{-3}$ g
Docking I/F	Magnetic	Under Development

Table 1. Key Parameters of the AMPHIS Prototype Spacecraft Simulator

1. Mass Reduction and Modular Construction

The first key enhancement is in the type of materials chosen for the structure. Prefabricated 6105-T5 aluminum fractional t-slotted extrusions form the cage of the vehicle while one square foot, .25 inch thick static dissipative rigid plastic sheets provide the form the upper and lower decks of each module. These materials provide the spacecraft simulator with high strength to weight ratio, ease of assembly and simple reconfiguration. Furthermore, their properties support the modular construction

technique taken for the simulator that allows for quick integration of different sensor packages and component troubleshooting.

The modular layout of the spacecraft simulator is evident in Figure 6. The lower module houses the propulsion and power subsystems and consists of two 68 cubic inch air cylinders, three air pads, a mounting platform for the thrusters currently under development, two Lithium-Ion batteries, and a DC-DC converter. The middle module houses a miniature Single Gimbaled Control Moment Gyroscope (MSGCMG) assembly and will house a mechanical relay array for computer control of the thruster and floatation solenoids and the CMG motor. The upper module houses a Windows XP based Pentium III PC-104 computer for requisite sensor data processing, a MATLAB XPC Target based Prometheus computer for executing the control algorithms, a high performance fiber optic gyro and a dual axis accelerometer. The top deck of the upper module provides a mounting for the indoor Global Positioning System (iGPS) sensor and receiver package and a LiDAR sensor. The selection criteria and performance characteristics associated with each component are presented in proceeding chapters.

2. Air Supply System Improvement

The second key enhancement involves the propulsion and power subsystems. Two 68 cubic inch, 3000 PSI carbon fiber air cylinders support the air requirements of three 32 mm air pads for floatation and future thruster requirements for translation control. The low mass of the spacecraft simulator, small air pads and efficient air supply system translate into approximately 75 minutes of vehicle floatation. Proper determination of the air flow requirement for translation control is yet to be conducted after they are implemented.

Containing the propulsion and floatation subsystems entirely in the lower module reduces the amount of piping and thus the possibility of air leaks in the system as well as simplifies the recharging process. Specifically, a quick disconnect (QD) style fitting between the manifold for the air cylinders and the vehicle allows them to be quickly removed and refilled by the collocated MAKO 5404BA air compressor. This compressor, capable of providing up to 6000 PSI of pressure, is a recent addition to the SRL that provides significant cost and time savings over the previous contracted refilled

cylinder replacement procedure. The procedures for the air supply refilling processes in the SRL are provided in Appendix C.

3. Improved Actuators for Attitude and Translation Control

The third key enhancement is in the actuator selection. Two new actuator designs for attitude and translation control are developed in order to improve the endurance of the simulator and test new concepts in small spacecraft motion in the POSF. In the case of the actuator providing translational motion, a dual rotating nozzle design is considered. This design provides the ability to correct for any center of gravity offsets from the center of mass, enhance the rotation rate of the vehicle if required and extend the endurance for the simulator by reducing the number of nozzles required provide full control in the two degrees of translation freedom. Full development of this actuator will be presented in Ref. [17].

For rotation torque about the one degree of attitude freedom in the POSF, a MSGCMG design is presented. This actuator, fully developed in the proceeding chapters, weighs just over one kg and provides a maximum torque of .334 N-m with an angular momentum of $49.4x10^{-3}$ N-m-s and peak power requirement of .02 W.

C. FLOATATION SYSTEM

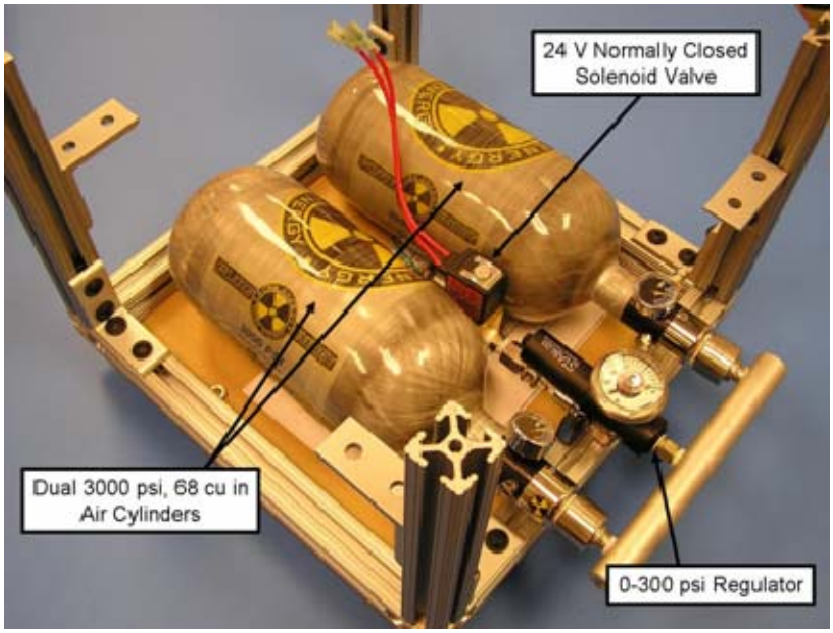

Figure 8. Floatation System Components in Lower Module

Figure 8 shows the component layout in the AMPHIS simulator. This system consists of the two 68 cubic inch, 3000 PSI spun carbon fiber tanks, a dual manifold and regulator, brass 1/8 inch National Pipe Tapered Thread (NPT) fittings, 1/8 inch Outer Diameter (OD) by 1/16 Inner Diameter (ID) tubing, one solenoid and three 32 mm air pads. The air from both tanks simultaneously feeds the system through a dual manifold and regulator assembly manufactured by Palmers Pursuit Shop capable of regulating between 0 and 300 PSI. The three 32 mm air pads, made by AeroDyne Belgium, are mounted on the underside of the first module in an equilateral triangle configuration to optimize their effect on the system. Tests conducted on the current configuration yield a flotation system air supply requirement of a minimum of 40 PSI in order to maintain a sufficient air gap of 10 micron between the pads and the floor. This air supply requirement translates into a total flow rate of 1.8 cubic inches per minute for the system, providing approximately 75 minutes of floatation. Figure 9 illustrates the floatation system in schematic form.

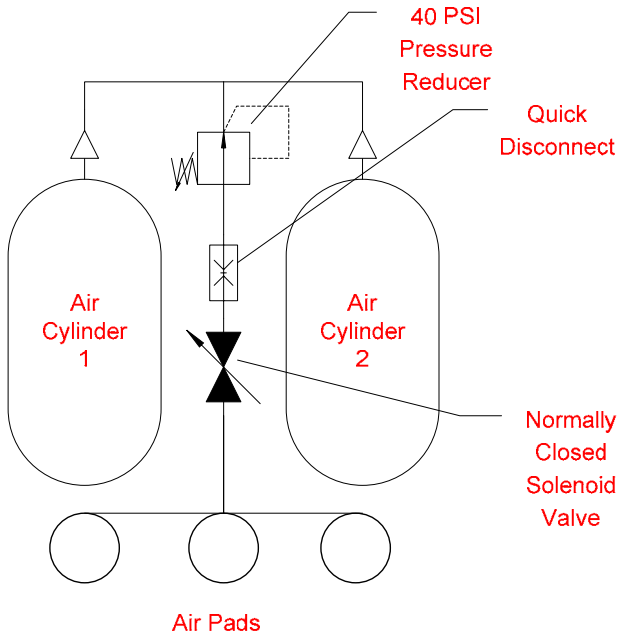

Figure 9. Floatation System Schematic

D. POWER DISTRIBUTION SYSTEM

Figure 10 shows the core components of the power distribution system mounted in the lower module of the AMPHIS simulator. Dual Lithium-Ion batteries by Ultralife Batteries, Inc, wired in parallel, provide 28 Volts for up to 12 Amp-Hours. These batteries are connected through a manual toggle switch to the input terminals on a Vicor Corporation four output terminal VIPAC DC-DC converter array. Each module houses components in the power subsystem and thus quick disconnects are installed between modules to provide rapid assembly and disassembly of the vehicle.

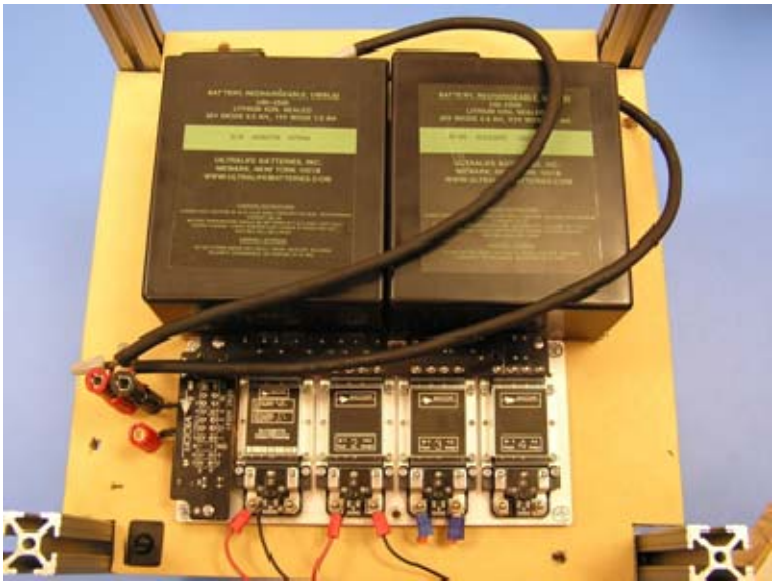

Figure 10. Power Distribution System Components in Lower Module

1. DC-DC Converter

A Vicor Corporation, four output terminal VIPAC DC-DC converter array is mounted in the lower module of the AMPHIS simulator (Figure 10). This array provides an easily modifiable solution for potentially changing component power requirements. The current configuration consists of one 20 Watt (5 Volts at 4 Amps) and three 100 Watts (24 Volts at 4.2 Amps) based on the current component power requirements listed in Table 2. For future requirement iterations, the outputs of the DC-DC converter can be trimmed up 10% and down 90% with some efficiency degradation trade-off. Given the large heat output of the converter array, an attached cold plate provides sufficient and unobtrusive heat transfer from the array to the AMPHIS power subsystem mounting deck.

2. Mechanical Relay Array

A Real Time Devices (RTD) Mechanical Relay Array will provide the requisite relay enabling control of the floatation and two thruster solenoid valves and the MSGCMG rotor wheel motor. The relay array has eight electromechanical single-pole, double throw relays that are driven by the digital output lines on the XPC Target based Prometheus PC-104 CPU. When commanded by the Prometheus CPU, the relay closes the appropriate circuit, allowing the requisite current to flow to the desired load. The relay requires +5 VDC, which is provided directly from the vehicle's bus.

3. Component Power

Table 2 summarizes the power requirements for the current and envisioned components on the AMPHIS spacecraft simulator. Presently, all components are wired directly to the DC-DC converter array although this is not the desired or practical arrangement for the final configuration. When installed, an electromechanical relay board will connect the control computer to the solenoid valves for the floatation and propulsion subsystems and thus will be wired between the DC-DC converter and the individual loads. Figure 11 illustrates the power distribution in schematic form.

Component	**Voltage Reqs (Max Power)**	**Component**	**Voltage Reqs (Max Power)**
Prometheus PC-104 CPU	5 [V] (5.5 [W])	SICK LD-OEM LiDAR	24 [V] (36 [W])
Versalogic EPM-CPU-10	5 [V] (27 [W])	ASCO Solenoid Valves	24 [V] (24 [W])
Netgear 4 Port Router	12 [V] (12 [W])	KVH DSP-3000 Fiber Optic Gyro	5 [V] (3 [W])
Maxon Motor EPOS 24/1 Encoder	9 [V] (18 [W])	Crossbow CXL02TG3 Accelerometer	5 [V] (10 [mW])
Super Precision Gyro Motor	5 [V] (5 [W])	Metris iGPS	Own Power Source

Table 2. Power Requirements on the AMPHIS Prototype Spacecraft Simulator

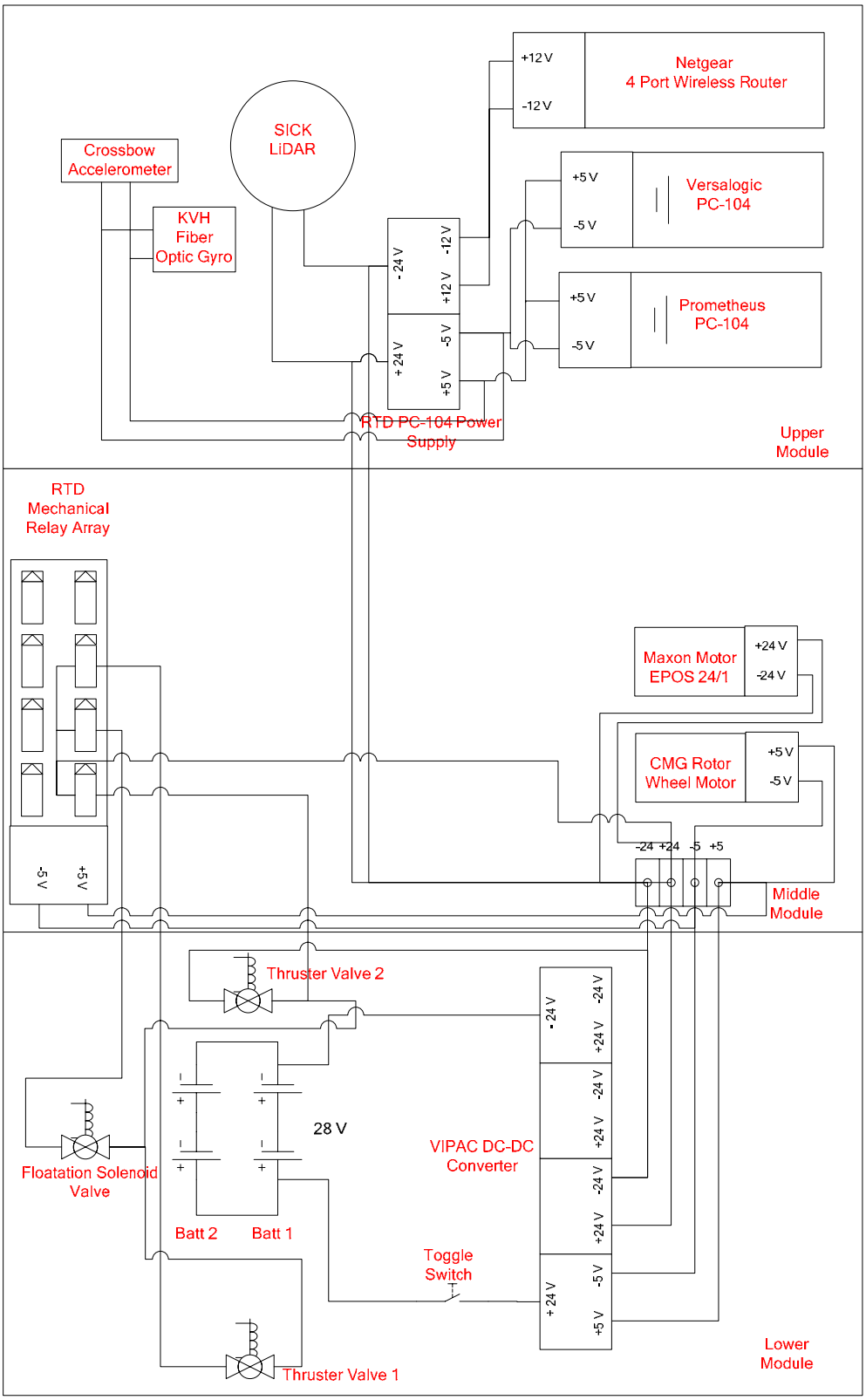

Figure 11. Power Distribution Schematic

E. PROPULSION SYSTEM AND ATTITUDE DETERMINATION AND CONTROL SYSTEM (ADCS) ACTUATORS

The simulator design includes actuators to provide both translational control for the propulsion system and rotational control for the ADCS. The actuating system for translational control is still under development but conceptually consists of dual rotating nozzles mounted on the same deck as the DC-DC converter array and batteries in the lower module. The nozzles are MJ5 nozzles made by Silvent. These nozzles are selected due to their previous documented superior performance at low pressure on the AUDASS system. (Ref. [16]) PreDyne EH2012-C204 solenoids are chosen over the ASCO model used for the floatation system due to their rapid switching capability of 3 to 5 milliseconds. Brushed DC motors and encoders by Maxon Motors are chosen to provide the required directional control for each thruster due to their high precision and equivalent power and computer interface requirements to those for the MSGCMG. A full development of the employed thruster system will be provided in Ref. [17].

1. Hardware Selection for the Attitude Control Actuator

A nominal slew rate of one deg/s is used in initial sizing of the MSGCMG for attitude control on the vehicle. This is a typical slewing requirement for small satellites. (Refs. [8],[9]) Furthermore, this slewing requirement concurs with the typical requirements experienced with the AUDASS testbed and can conceptually be improved if necessary by the addition of a scissor-type dual MSGCMG system. (Ref. [13])

Figure 12 shows the MSGCMG assembly mounted in the second module of the AMPHIS vehicle. A Super Motorized Precision Gyroscope by Educational Innovations, Inc. is at the core of the assembly. The gyroscope is rotated on the plane of the spinning rotor wheel by a graphite brush DC motor, commanded through an encoder both manufactured by Maxon Motor. A RS-232 interface provides the hardware interface between the control computer and a Maxon Motor EPOS 24/1 digital motion controller. Due to the current design requirement to use an MATLAB XPC Target based control computer for its real-time processing capability, requisite software to communicate with the EPOS 24/1 required development and is presented in Appendix D. The employed communication protocol is outlined in section three of this chapter.

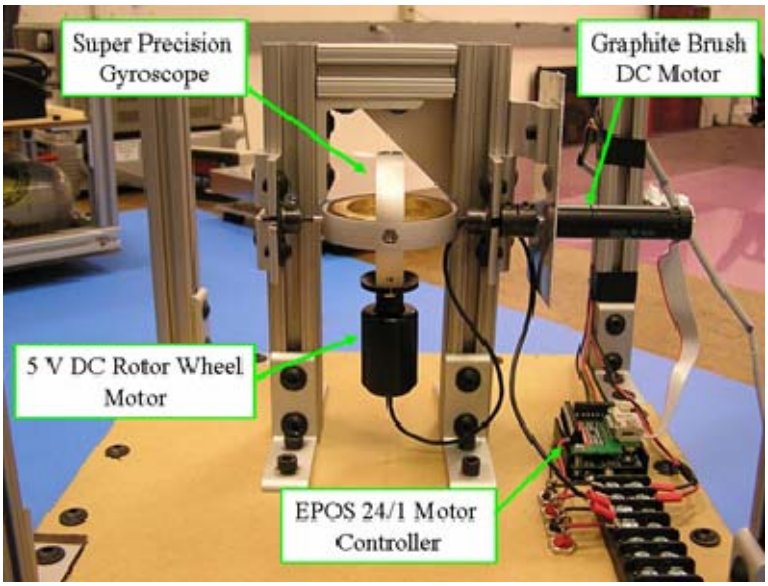

Figure 12. MSGCMG Assembled in Second Module of AMPHIS

To determine the key characteristics of the MSGCMG such as the true maximum torque applied by the gimbal motor and its maximum angular rate, several measurements were taken using the EPOS User Interface software provided by the manufacturer. Figure 13 shows the EPOS User Interface produced graph of the position, velocity and current values during a 90 degree slew maneuver. The maximum gimbal angular rate $\left(\dot{\delta}\right)$ was determined to be 6.95 rad/s by commanding the gimbal motor using the position profile from the graph. Similarly, the maximum torque of the gimbal motor was determined to be $4.06x10^{-3}$ N-m $\left(T_g\right)$ by analyzing the current profile through the maneuver and taking the product of the max current and the torque constant (8.11 Nm/A). This determined torque validates the advertised maximum continuous torque of $4.98x10^{-3}$ N-m. Finally, the moment of inertia for the gimbal assembly $\left(J_g\right)$ was determined by taking several experimental measurements of the time taken for the gimbal motor to complete a 90 degree maneuver and integrating the equation of motion for torque about one axis yielding

$$J_g = \frac{T_g \Delta t^2}{2\Delta\theta} = 37.17x10^{-6} \text{ kg-m}^2 \quad (1)$$

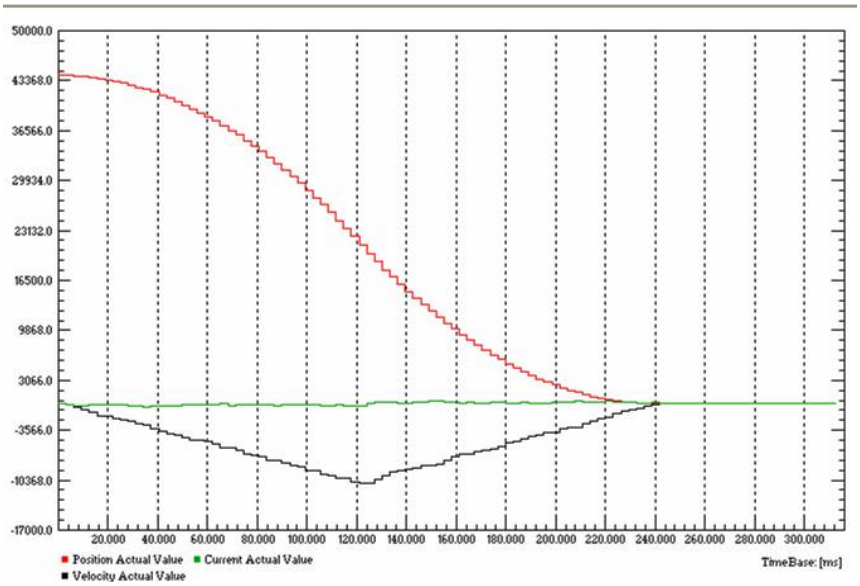

Figure 13. Postion, Velocity and Current History for a 90 deg Gimbal Motor Turn

With the maximum gimbal motor torque $\left(T_g\right)$ and the gimbal moment of inertia $\left(J_g\right)$, the actual maximum angular acceleration $\left(\ddot{\delta}\right)$ of the gimbal motor was determined to be 109.9 rad/s^2.

The moment of inertia about the rotating axis for the rotor wheel was calculated using the theoretical moment of inertia for a flat disk as

$$J_w = \frac{1}{2} m_w r_w^2 = 3.717 x 10^{-5} \text{ kg-m}^2 \tag{2}$$

where m_w and r_w are the mass and the radius of the rotor wheel respectively. Using this and the advertised rate of rotation for the rotor wheel of 12000 rpm, the angular momentum of the rotor wheel is

$$h_w = J_w \omega_w = 49.4 \text{ mN-m-s} \tag{3}$$

Given the angular momentum of the wheel from Eq. (3) and the maximum gimbal rate of the motor, the maximum torque capable from the MSGCMG is

$$T_{CMG}(\max) = h_w \dot{\delta}_{\max} = 344 \text{ mN-m} \tag{4}$$

These values are used to develop both the simulation and experimental models for the maneuver about the z-axis presented later and are summarized in Table 3 along with other key technical specifications of the MSGCMG.

Rotor Wheel	
Moment of Inertia Wheel (J_w)	$3.717x10^{-3}$ [kg-m^2]
Maximum Momentum Wheel (h_w)	$49.4x10^{-3}$ [N-m-s]
Maximum Wheel Speed	±1256.6 [rad/s] (12000 rpm)
Power Supply	5 [VDC]
Gimbal Motor & Encoder	
Maximum Gimbal Rate	6.95 [rad/s] (398 deg/s)
Maximum Gimbal Acceleration	109.9 rad/s^2 (6297 deg/s^2)
Maximum Gimbal Torque	$4.06x10^3$ [N-m]
CMG	
Total Mass of CMG (including mounting hardware)	1.148 [kg]
Power Supply	9-24 VDC
Maximum Output Torque	.344 [N-m]
Power	< 24 [W]
Interface	RS-232
Dimensions	.02 [m^2]

Table 3. Key Technical Specifications for the AMPHIS MSGCMG

2. Application for Single Axis Slew Maneuver

The utility of the MSGCMG for the floating three DOF spacecraft simulator can be shown through development of the equations of motion for the vehicle and the CMG. Beginning with the well developed equation of motion for a rigid spacecraft with a CMG (Ref. [18])

$$\dot{\mathbf{H}}_s + \boldsymbol{\omega} \times \mathbf{H}_s = \mathbf{T}_{ext} \tag{5}$$

where $\mathbf{H}_s$ is the angular momentum vector of the total system, $\mathbf{T}_{ext}$ is the external force vector and $\boldsymbol{\omega} = \left[\omega_x, \omega_y, \omega_z\right]^T$ is the angular velocity vector of the vehicle. All vectors are body-fixed. Figure 14 illustrates the vector relationships in the body frame with the z-axis being the vertical axis of the vehicle, the x-axis aligned with the gimbal motor shaft and the y-axis completing the right-handed coordinate system.

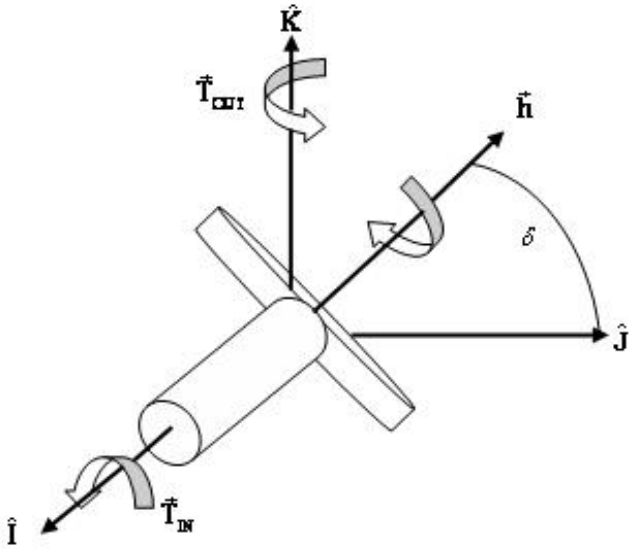

Figure 14. Vector Relationships Within the SGCMG

The full angular momentum vector $\mathbf{H}_s$ takes into account both the main vehicle angular momentum and the MSGCMG angular momentum so that

$$\mathbf{H}_s = \mathbf{J}\boldsymbol{\omega} + \mathbf{h} \tag{6}$$

where $\mathbf{J}$ is the moment of inertia matrix of the vehicle including the MSGCMG and $\mathbf{h} = \left[h_x, h_y, h_z\right]^T$ is the total MSGCMG momentum vector. Differentiating Eq. (6) to find $\dot{\mathbf{H}}_s$ yields

$$\dot{\mathbf{H}}_s = \dot{\mathbf{J}}\boldsymbol{\omega} + \mathbf{J}\dot{\boldsymbol{\omega}} + \dot{\mathbf{h}} \tag{7}$$

Combining Eqs. (5), (6) and (7), and assuming no external forces due to the vehicle being maintained in a nearly frictionless and gravity free environment with its air pads and the flat floor in the POSF yields

$$\mathbf{J}\dot{\boldsymbol{\omega}} + \dot{\mathbf{J}}\boldsymbol{\omega} + \dot{\mathbf{h}} + \boldsymbol{\omega} \times (\mathbf{J}\boldsymbol{\omega} + \mathbf{h}) = \mathbf{0} \tag{8}$$

The change in the total moment of inertia is negligible in the vehicle due to the size of the MSGCMG and the use of compressed air in the propulsion system. Introducing the torque produced by the MSGCMG denoted as $\mathbf{T}_{\text{CMG}} = \left[T_x, T_y, T_z\right]^{\text{T}}$ and a control vector $\mathbf{u} = \left[u_x, u_y, u_z\right]^{\text{T}}$ to represent a required torque from a chosen control law, Eq. (8) becomes

$$\mathbf{J}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} = \mathbf{T}_{\text{CMG}} \tag{9}$$

where

$$\mathbf{T}_{\text{CMG}} = -(\dot{\mathbf{h}} + \boldsymbol{\omega} \times \mathbf{h}) = \mathbf{u} \tag{10}$$

Referring to Figure 14, the MSGCMG angular momentum vector is

$$\mathbf{h} = h_w \left[0, \cos\delta, \sin\delta\right]^{\text{T}} \tag{11}$$

where h_w is the constant angular momentum of the rotor wheel. By considering only the component along the z-axis of the spacecraft simulator (i.e. $\boldsymbol{\omega}_z, \mathbf{I}_z, \mathbf{T}_z$, etc.) since the horizontal components are compensated by the reaction of the floor, Eqs. (5) and (6) simplify to

$$\mathbf{J}_z\dot{\boldsymbol{\omega}}_z = \mathbf{T}_{\text{CMG}_z} = -\dot{\mathbf{h}}_z = \mathbf{u}_z \tag{12}$$

where

$$\dot{\mathbf{h}}_z = h_w \cos\delta\dot{\delta} = -\mathbf{u}_z \tag{13}$$

To determine a steering logic for the instantaneous angular rate to command the gimbal, it is necessary to solve Eq. (13) for the gimbal rate $\dot{\delta}$ given a required torque produced by the attitude control law. For initial modeling and testing of the MSGCMG, a standard PD control is considered both for the control of the gimbal motor and the vehicle's rotation due to the simple nature of the attitude control. Simulation and experimental results are presented in Chapter III with the models used in SIMULINK represented in Appendix D.

3. Communication Protocol for MSGCMG

As mentioned previously, serial RS232 communication protocol is used for transmitting and receiving data between the control computer and the EPOS 24/1 positioning controller. This protocol uses the RS232 standard for transmitting and receiving data over a three wire combination of TxD, RxD and GND. This wiring configuration corresponds to the 2, 3 and 5 pins respectively on the RS-232 interface. The data is transmitted asynchronously with one start bit, eight data bits, no parity and one stop bit. Each data byte is transmitted in sequential frames with each frame consisting of a header, a variably long data field and a 16-bit cyclic redundancy check (CRC) to verify data integrity, all in hexadecimal format. The header is a one word (16-bit) value composed of an 8-bit operation command (OpCode) followed by an 8-bit value that represents the number of words in the data field minus one. The data field contains the parameters of the particular message and is separated into groups of 16-bit words. The final component of the frame structure, the CRC, is a 16-bit word that must be calculated before transmitting using the supplied C code provided by the manufacturer and then converted to MATLAB code for use in MATLAB. In calculating the CRC, the code looks first at the header values followed by the high byte and then the low byte of each sequential data value followed finally by 0x0000. Before transmitting, the string must be reorganized to place the low bytes first followed by the high byte for each data word and the CRC word. (Ref. [19]

The necessary software to control the MSGCMG through the EPOS 24/1 controller in MATLAB did not exist and therefore required development. The EPOS User Interface Software provided by the manufacturer enabled rapid determination of the header and data values for the necessary functions such as clearing any faults, setting the different PID controller gains and the send desired position and velocity values to the motor. These header and data values are sent through a function that computes the CRC and then are prepared and transmitted. The developed MATLAB functions are provided in Appendix D.

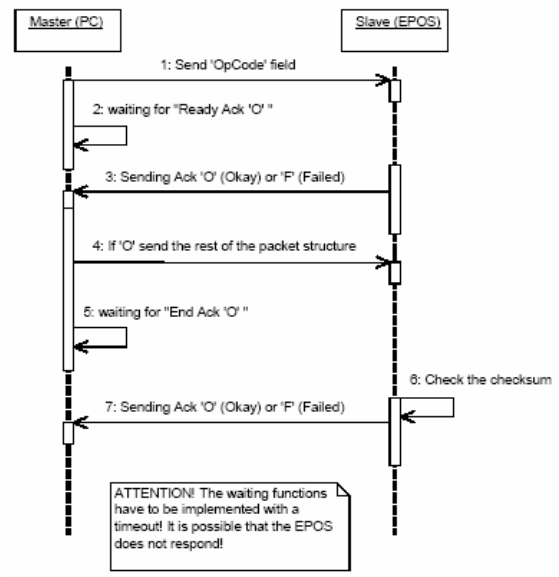

Figure 15. Interaction Diagram for Sending a Frame Structure (Ref. [19])

The communication rate with the EPOS 24/1 controller is limited to 1 kHz for positioning and velocity control but increases to 10 kHz for current control. Figure 15 shows the interaction diagram for sending a frame structure to the EPOS. In the developed function, a check to verify that bytes are in the buffer is implemented before reading data and if not a brief pause is initiated to allow for data to be available.

F. SENSORS

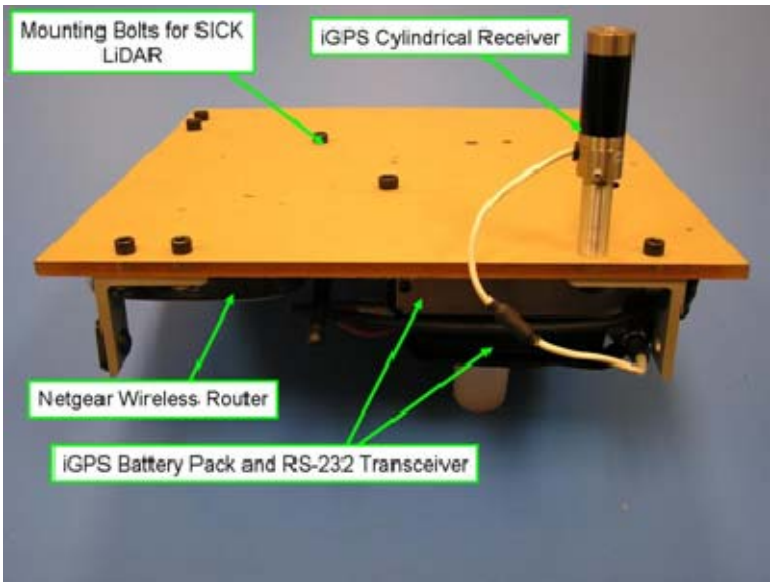

Figure 16. Component Layout of the Sensor Deck

Figure 16 shows the sensor deck which forms the top of the upper module. The sensor deck houses all the components that provide the interface between the simulator and its environment with the exception of the inertial measurement devices which are attached to the bottom deck of the upper module.

The required components to provide the simulator with precise position information through the iGPS including a receiver, a RS-232 transmitter and battery pack are mounted in order to optimize their use. A wireless 4 port router by Netgear is mounted on the underside of the deck to provide onboard Ethernet connection to both on-board computers. Finally, a SICK Laser Scanner is mounted in the center of the deck with three bolts.

1. Laser Scanning Sensor

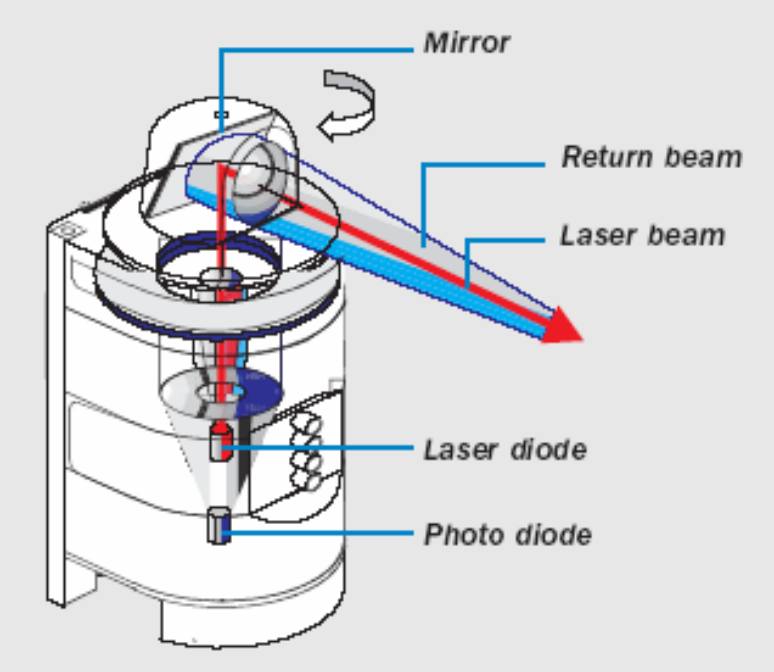

Figure 17. SICK LD-OEM Laser Scanner (Ref. [20])

The primary sensor for determining relative attitudes and position of the prototype AMPHIS simulator to the other simulators is the SICK LD-OEM Laser Scanner. Figure 17 illustrates the basic configuration of the scanner. The mirror assembly on the top of the scanner rotates about the –Z axis of the simulator, providing range and bearing information from the resolved outgoing and incoming laser pulses in the X-Y plane. The

scanning frequency is between 5 and 20 Hz with a maximum scan rate of 14.4 kHz. The system weighs 3.2 kg and is 115 x 120.5 x 222 mm in dimension. A best case angular resolution of .125 degrees is possible with a maximum range of 24 m for a 5% reflectivity object, and 100 m for a 90% reflectivity object. Communication is established between the control computer and the LD-OEM via RS-232. Full development of the requisite communication protocol using MATLAB XPC Target and extracted information will be provided by Ref. [21]. Figure 18 shows a sample of the typical scene received from the laser scanner with the dots representing a positive return.

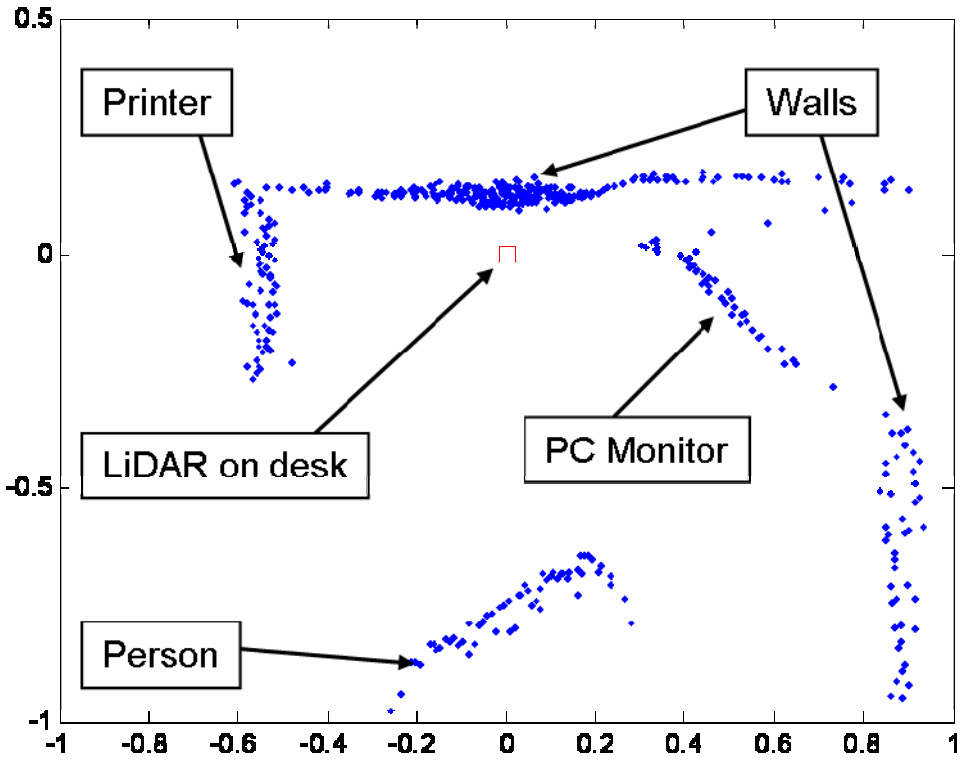

Figure 18. Typical Scene Results from SICK LD-OEM

2. Indoor Global Positioning System (iGPS)

There is great potential for the use of Global Positioning System measurements for position and velocity determination in future on-orbit spacecraft assembly missions. By using carrier-phase differential GPS for Low Earth Orbiting (LEO) satellites, position measurement accuracy on the order of 2-5 cm can be achieved. (Ref. [22]) The POSF is

equipped with an iGPS by Metris. This system is similar to the on-orbit GPS system in that it is capable of providing high precision position determination. However, unlike the use of radio signals by orbiting GPS satellites, the iGPS system uses laser transmitters to cover the testbed with infrared light that is detected and then processed using on-board software. (Ref. [23]) Figure 16 shows the iGPS receiver mounted on the sensor deck of the simulator. In addition to the receiver on the top of the sensor deck, a battery pack and RS-232 interface pack are mounted in its current configuration. This system is capable of providing ~1 mm accuracy at a range of between 2 and 40 m. For this reason, two transmitters are mounted 2.5 m off the ground attached to the lab walls at a distance of 4 m from the floor and 5 m apart. A full discussion of the mechanics of the iGPS system and its setup is available in Ref. [15]. Additionally, the procedures for setting up the necessary Windows based software on the Versalogic PC-104 to process the received information on-board the AMPHIS simulator is provided in Appendix C. Once processed, the position coordinates are sent from the Versalogic PC-104 to the MATLAB XPC Target based Prometheus PC-104 control computer for use by the control algorithm.

3. Inertial Measurement Devices

A KVH fiber optic gyro and a Crossbow three-axis accelerometer provide inertial measurement requirements for the spacecraft simulator. The KVH DSP-3000 Fiber Optic Gyro selection is based on its compact, lightweight frame, its low bias term (see Table 1) and its serial interface. Similarly, the Crossbow CXL02TG3 accelerometer, with a bias stability of $\pm 8.5x10^{-3}$ g, is equipped with a serial interface allowing for rapid integration into the existing control system architecture. Full development of the applicable communication protocols for both devices will be provided in Ref. [17]. Both measurement devices are mounted in the upper module to avoid interference vibration producing sources and allow direct connection through a Diamond Systems Digital/Analog board for the accelerometer, and RS-232 connection for the fiber optic gyro, to the Prometheus control computer.

G. COMMAND AND DATA HANDLING SYSTEM

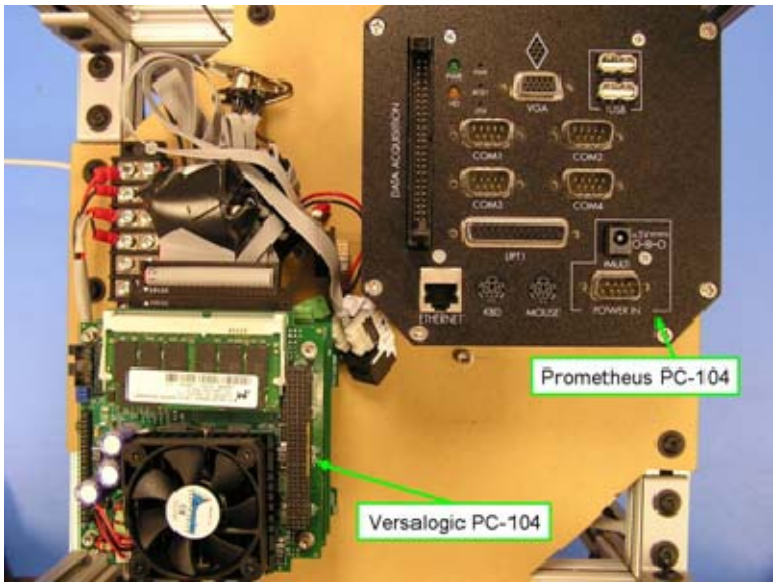

Figure 19. Dual PC-104 Computers Mounted in the Upper Module

Figure 19 shows the AMPHIS simulator's two PC-104 computers mounted in the upper module. A Diamond Systems Prometheus High Integration PC/104 computer, running the MATLAB XPC Target Operating System, delivers real-time control capability to the simulator while a Windows XP based Versalogic Jaguar Pentium III, EPM-CPU-10 processes the raw Laser Scanner and iGPS information into useable data and then sends this processed data to the Prometheus control computer. The size, capability and rapid upgradeability of embedded PC-104 computers drove their selection over other types of CPUs. Both PC's are equipped with Ethernet and RS-232 serial ports used to connect them to each other through the router and to the pertinent sensors respectively. A key advantage of PC/104 computers is that they contain both an ISA (Industry Standard Architecture) and PCI (Peripheral Component Interface). (Ref. [15])

The Prometheus PC includes a full interface board with support for up to 4 RS-232 serial devices, data acquisition via a digital/analog board, Ethernet, dual USB (Universal Serial Bus) ports, LPT (Line Printing Terminal) and KVM (Keyboard-Video-Mouse) interfaces. Additionally, power is supplied to the computer through a +5 VDC connection on the I/O board. In its current configuration, all processes are run via a 32 MB flashdisk containing the boot code for MATLAB XPC Target on its 486-DX2

processor with 100MHz co-processor. Due to the lack of support for the pre-configured Ethernet adapter on the Prometheus I/O board, a fully compatible Winsystems PCM-NE2000 IEEE 802.3 Ethernet Controller PC104 Module is placed at the bottom of the stack for TCP/IP (Transmission Control Protocol/Internet Protocol) communication. Appendix B includes the procedures for setting up the computer.

The Versalogic Jaguar PC is based on the Windows XP operating system running on a 850 MHz Pentium III processor with 256 MB DRAM (Dynamic Random Access Memory). It is equipped with KVM, dual USB, Ethernet and dual RS-232 communication ports as well, supporting either standard or laptop sized IDE disk drives and a 3.5 inch floppy drive. Additionally, an Embedded Designs Plus firewire board is installed to support CMOS (Complementary Metal Oxide Semiconductor) camera sensor capability if desired. MATLAB is installed to support initial testing of the MSGCMG and other sensors onboard the simulator. The iGPS software "Workspace" is installed to process the raw data from the transmitters in the POSF through the onboard router into position coordinates. The procedure to setup the iGPS software is provided in Appendix C.

H. SUPPORT EQUIPMENT

1. Air Charging Station

Figure 20. CompAir MAKO 5404BA Layout in the SRL

A key improvement to the POSF comes in the refurbishment of an existing compare MAKO 5404BA air compressor. This air compressor alleviates the need for the costly and time consuming practice of continuous replenishment of the large air cylinders by outside vendors as was the case in previous years. Figure 20 shows the compressor in its current configuration in the SRL. The compressor features a 5 HP motor that is capable of producing breathing quality air through three compression stages up to 5000 PSI (345 Bar). The unit fills an attached 6000 PSI air tank equipped with an adjustable regulator and two whips. The first whip is configured to fill the legacy AUDASS systems 4500 PSI capable SCBA (Self Contained Breathing Apparatus) air cylinder while the second whip is configured to fill the 3000 PSI capable air cylinders via a QD style fill fitting for the AMPHIS simulators. Procedures for operating the air compressor and refilling the air cylinders are provided in Appendix D.

2. Battery Recharging

The vehicle's dual lithium batteries are recharged by simply detaching them from the connecting whips in the lower module and then attaching them to the manufacturer provided battery re-charger. The full procedure for charging the battery packs is detailed in Appendix D. A total recharge takes approximately 3 hours and each battery has its own charger. Additionally, if the batteries are left attached to the charger, it will initiate a low rate charge every 24 hours to maintain a fully charged state.

THIS PAGE INTENTIONALLY LEFT BLANK

III. PERFORMANCE

A. PERFORMANCE RESULTS OF ONE AXIS MSGCMG FOR THE AMPHIS SPACECRAFT SIMULATOR

In determining whether the MSGCMG can perform as required and to develop the control code required to command the EPOS 24/1, initial modeling was conducted using MATLAB and SIMULINK. The dynamics of the gimbal are modeled according to the one axis of rotation equation of motion

$$T_g = J_g \ddot{\theta} \tag{14}$$

where T_g is limited to the experimentally derived value from Figure 13 and J_g is taken to be the derived value from Eq. (1). A Proportional-Derivative (PD) control law is applied to the gimbal motor system to minimize the error between the reference gimbal rate and gimbal angle produced using the derived steering logic from Eq. (13)and the true gimbal rate and angle from the motor's plant. Initial gain selection is based on the equation of motion governing the rotation of a rigid body about one axis

$$\Delta\ddot{\theta} + \left(2\xi\omega_n\right)\Delta\dot{\theta} + \left(\omega_n^2\right)\Delta\theta = 0 \tag{15}$$

where ω_n is the natural frequency of the system and ξ is the damping ratio. The natural frequency can be approximated by examining the attitude control system based on proportional-derivative control in Figure 21 where the torque is directly proportional to the angle of rotation.

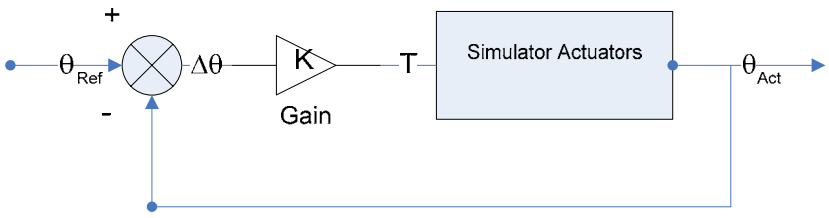

Figure 21. Proportional-Derivative Control Attitude Control System

Therefore, the error dynamics can be represented as

$$\Delta\ddot{\theta} + \left(\frac{K_D}{J}\right)\Delta\dot{\theta} + \left(\frac{K_P}{J}\right)\Delta\theta = 0 \tag{16}$$

and by looking only at the first-order response, the proportional gain for the controller can be determined. Assuming a rigid, homogenous rectangular prism structural characteristic with the dimensions of the vehicle listed in Table 1 and the parameters of the MSGCMG listed in Table 3, the proportional gain for the simulator controller considering a 10 degree maneuver is

$$K_p \cong \frac{T_{MAX}}{\Delta\theta} = \frac{.3435}{.1745} = 1.97 \text{ Nm/rad} \tag{17}$$

Now, by equating the coefficients of Eq. (15) and Eq. (16) and taking the damping ratio to be .9 (typical of a critically damped system), the natural frequency of the simulator is estimated as

$$\omega_n = \sqrt{\frac{K_p}{J_z}} = 1.518 \text{ rad/s} \tag{18}$$

and therefore,

$$K_D = 2J_z\xi\omega_n = 2.416 \text{ Nm/(rad-s)} \tag{19}$$

A similar process is conducted to determine the starting gains for the gimbal motor controller. After initial gain determination, the gains for both controllers are tuned to yield proper results for a critically damped system.

1. Simulation Results

Results for a 30 degree maneuver shown in Figure 22 demonstrate the designed MSGCMG is capable of completing the maneuver (within .001 rad, .01 rad/s and .01 rad/s^2) in 55.8 sec or a .5 deg/s slew with a worst case moment of inertia of 1.5 kg-m^2. Figure 23 demonstrates that a 2 deg/s slew can be accomplished on a better estimate of the true moment of inertia for the spacecraft simulator of .25 kg-m^2 from experimental results. This potential capability is confined by the experimentally derived values for the maximum gimbal motor output torque and angular rate and assumes no friction in the rotor wheel or gimbal delays. Additionally, a perfectly external torque-free environment and a homogenous rigid structure are assumed. Further tuning of the PD control and/or

changing control laws could potentially increase the slew rate of the maneuver and will be investigated in further hardware-in-the loop experimentation.

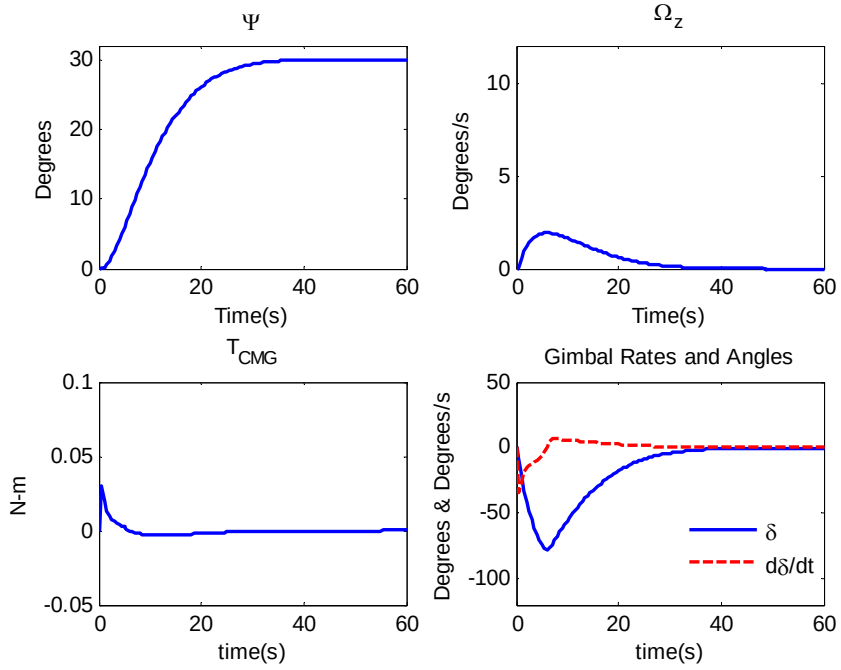

Figure 22. 30 degree Rest-to-Rest Slew Maneuver ($J_z = 1.5$ kg-m^2)

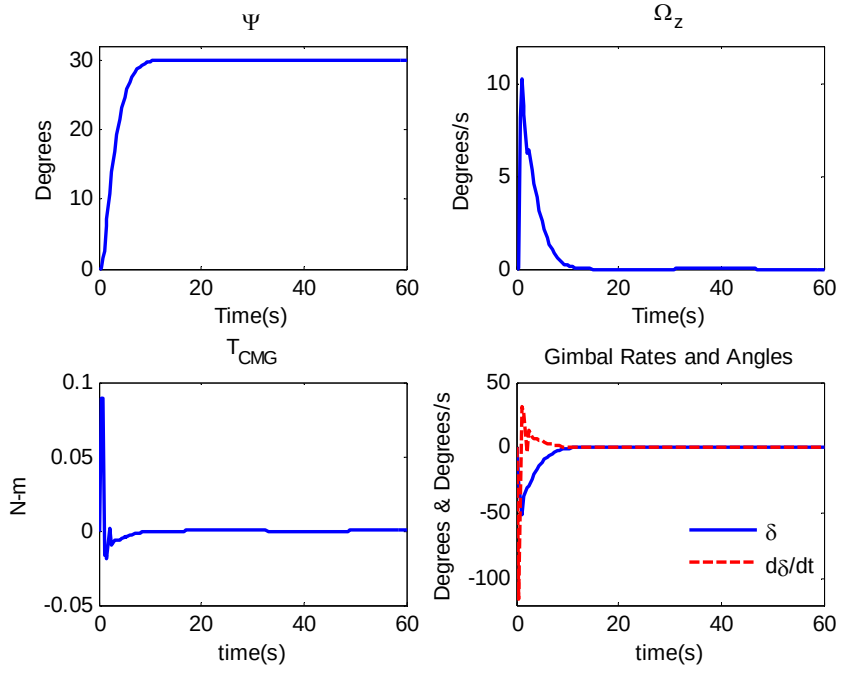

Figure 23. 30 degree Rest-to-Rest Slew Maneuver ($J_z = .25$ kg-m^2)

2. Preliminary Experimental Results

Angular readings from the fiber optic gyro for an open-loop test of the MSGCMG on the AMPHIS prototype spacecraft simulator using MATLAB running on the Windows XP based CPU are shown in Figure 24. For this test, the gimbal rates and gimbal angles determined from the simulation for a $J_z = .25$ kg-m^2 were sent to the vehicle using the commands in Appendix D. These results demonstrate an improvement in the slew rate (4.8 deg/s) over the simulated results. However, by using the results of the closed-loop simulated system in an open-loop experiment, the vehicle travels over the desired final angle by six degrees. This overshoot is due primarily to un-modeled disturbance forces to include gimbal delays, friction in the motor and estimated moment of inertia. Further development of the necessary code to command the EPOS 24/1 in real-time is required along with the necessary code to include the data provided by the rate gyro. Once completed, additional tests will be conducted to validate a closed-loop system response.

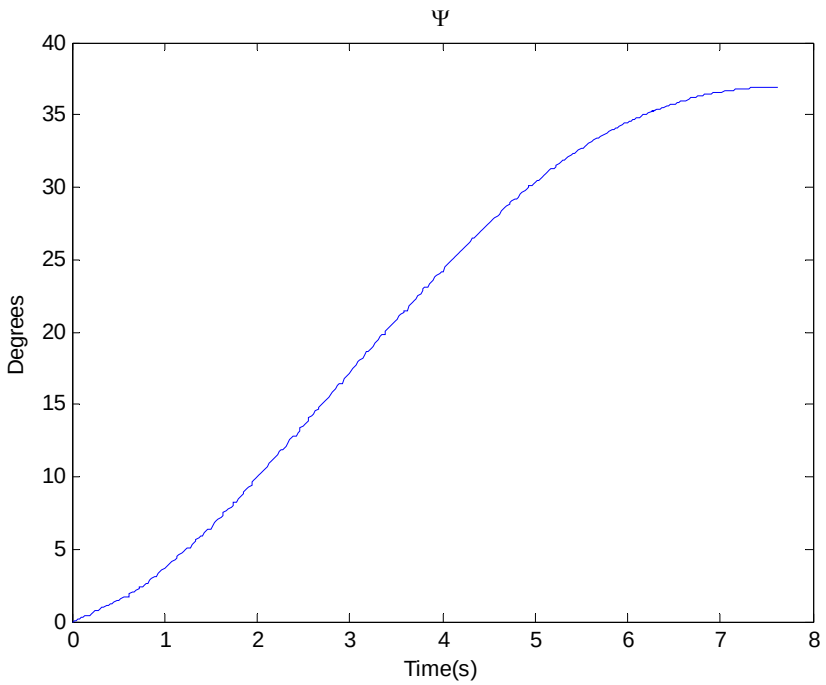

Figure 24. Experimental Open Loop 30 degree Rest-to-Rest Slew Maneuver

B. PROPELLANT AND BATTERY EFFICIENCY

Initial experimentation has been limited to testing of the MSGCMG and thus the only draw on the air cylinders has been from the floatation system. In this limited

configuration, there has been approximately 75 minutes of continuous time available for experimentation. By using the data available for the AUDASS system which had a total of eight thrusters and a significantly larger platform to support with its floatation system, a maximum air consumption value can be determined and then a fair approximation of the realistic propellant consumption for the AMPHIS vehicle can be estimated.

During initial testing, the AUDASS vehicle used approximately 200-300 PSI per minute. (Ref. [16]) With the designed dual thruster combination and a vehicle of approximately a third the mass but half the air cylinder size, air consumption on the AMPHIS vehicle is estimated at 150 PSI at a maximum. With this value, it is estimated that the AMPHIS vehicle will be capable of operating for at least an equivalent time to the AUDASS vehicle of 15 minutes if not slightly longer.

In the case of battery efficiency, with the current configuration at least five hours of continuous use has been achieved. In future iterations, providing the minimum voltage to each component instead of a standard 24 VDC and 5 VDC bus may extend this further although, with the limiting factor being the air supply and given multiple batteries, there is little benefit in investigating additional power savings.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. CONCLUSION

A. SUMMARY

The objective of this thesis was to design and build a new prototype spacecraft simulator for the AMPHIS testbed. Throughout the design and integration process, the lessons learned during the component and construction phases of the AUDASS simulator were considered. In leveraging these lessons learned, three key enhancements were made to the prototype AMPHIS spacecraft simulator enabling its use for hardware-in-the-loop experimentation of potential GNC algorithms for on-orbit spacecraft assembly in the SRL's POSF. The selection of aluminum t-slotted components for the frame and rigid static dissipative plastic for the decking provide sufficient strength for the vehicle while allowing it to be rapidly assembled and reconfigured as necessary. A streamlined floatation system consisting of only three 32 mm air pads fed by two 68 cu in air cylinders through a compact and lightweight dual manifold and regulator system provides nearly 75 minutes of necessary floatation for experimentation. With the designed low mass, low power and compact MSGCMG, the vehicle is able to rapidly adjust its attitude with preliminary experimental results demonstrating a 4.8 deg/s slew rate.

B. FUTURE WORK

With the core conceptual design for the prototype spacecraft simulator complete, further development of several components is necessary to fully outfit the AMPHIS testbed. This component development currently includes the thrusters, mechanical relay, and all the onboard sensors presented in this thesis. Furthermore, after full construction of the prototype is complete, at least two more vehicles are needed to begin full-scale hardware-in-the-loop investigations into potential GNC algorithms for on-orbit spacecraft assembly using fractionated systems.

1. Thruster Design and Integration

As mentioned in this thesis, dual rotating thrusters are envisioned to provide the translational motion control for the AMPHIS spacecraft simulator. Each thruster will be driven by compressed air via a 24 VDC normally closed solenoid valve and their pointing will be controlled by the same type of motor, encoder and positioning controller used by the MSGCMG assembly. A key contribution of this thesis toward full integration of

these thrusters is in the area of the necessary software development to communication with the EPOS 24/1 via serial interface using MATLAB but further work needs to be accomplished to streamline this code. In general, a change from a Windows and MATLAB based architecture to a Linux and C based architecture could potentially speed up communication between all components and the onboard PC-104.

2. New Computer Architecture

The current demand for two PC-104s to operate the spacecraft simulator could be reduced to a single Linux based computer capable of providing near real-time processing after a few key achievements. First, the software which converts the raw data streaming from the onboard iGPS receiver into usable position coordinates is currently only Windows based. Development of Linux compatible software to provide this same type of data could potentially speed up communication with the receiver while eliminating the requirement for a Windows PC.

Additionally, with the rapid advancement of the processor speeds for PC-104 computers, the Versalogic Pentium III computer could be substituted with a significantly faster Pentium IV version. This substitution could allow for higher bandwidth communication with the sensor and actuators, increasing the update rate for potential GNC algorithms.

3. Increased Miniature CMG Applications

The inclusion of the MSGCMG on the AMPHIS spacecraft simulator is a unique attribute of the Spacecraft Robotics Laboratory and thus presents significant research potential. Efforts to reduce the friction in the rotor wheel and gimbal assembly and encase the assembly have the potential to enhance the MSGCMG's robustness and increase its reliability. Furthermore, the addition of a second MSGCMG in a scissor type configuration, or the placement of a full complement of four MSCGMG's in a tetrahedral configuration with a spherical air bearing support for the simulator's bus could be used to conduct research into the capabilities of CMGs for small satellite applications.

APPENDIX A. COMPONENT MANUFACTURER AND LIMITATION INFORMATION

A. STRUCTURAL COMPONENTS

Aluminum T-Slotted Framing System:

Manufacturer:	80/20
Vendor:	McMaster-Carr
	http://www.mcmaster.com/

Static Dissipative Rigid Plastic Sheets (12 x 12 x ¼ inch)

Vendor:	McMaster-Carr

B. FLOTATION SYSTEM

Air Cylinders:

Manufacturer:	Pure Energy paintball bottle, Model #40669
Vendor:	Palmers Pursuit Shop
	(http://www.palmer-pursuit.com)
Limitations:	3000 PSI, 68 cu in per
	QD Style fill fitting

Stabilizer Dual Manifold System:

Manufacturer:	Palmers Pursuit Shop
Limitations:	Dual paintball bottles
	Regulated pressure between 0 and 300 PSI

1/8 in OD- 1/16 in ID Low-Pressure Poly Tubing:

Vendor:	Palmers Pursuit Shop
Limitations:	500 PSI

1/8 NPT Brass Fittings (Various):

Vendor:	Palmers Pursuit Shop
Limitations:	500 PSI

Solenoid

Manufacturer:	ASCO, Model #U 8225B002V
Limitations:	24 VDC, Normally Closed, rated for fluid and gas Maximum pressure of 125 PSI

32 mm Air Bearings:

Manufacturer:	Aerodyne Belgium, PERARA Dextair (PE032)
Vendor:	Ameropean (No longer distributing)
Limitations:	Maximum Loading of 125 N @ 4 bar ensures 10 micron air gap

C. POWER DISTRIBUTION SYSTEM

Lithium Ion Battery Packs:

Manufacturer:	UltraLife Batteries, Inc., Model #UBBL02 http://www.ultralifebatteries.com/
Limitations:	28 V for 6 AH or 14 V for 12 AH per battery pack

DC-DC Converter Array:

Manufacturer:	Vicor, Standard VIPAC Array http://www.vicr.com/
Limitations:	Input Voltage 24 VDC Max power output per converter 100 W

Mechanical Relay Array:

Manufacturer:	RTD Embedded Technologies, Model # DMR8 http://www.rtdusa.com
Limitations:	±5 VDC operating Capable of supporting 8 separate devices

D. ACTUATORS

Thruster Solenoids:

Manufacturer:	Precision Dynamics, Inc., Model # EH2012-C204

http://www.predyne.com

Limitations: +24 VDC, Normally Closed, rated for fluid or gas 3-5 milliseconds switching capability

Thruster Nozzles:

Manufacturer: Silvent, Model # MJ5
http://www.silvent.com

Limitations: 5.9 scfm air consumption, 1.8 N force with 72 PSI supply, M5x.5 connection

MSCMG gyroscope:

Manufacturer: Educational Innovations, Inc., Super Motorized Precision Gyroscope
http://www.teachersource.com

Limitations: ±5 VDC, 12000 rpm rotor wheel rotation rate

MSCMG gimbal motor:

Manufacturer: Maxon Motor USA,
RE16 motor, Model # 118730
MR encoder, Model # 201940
EPOS 24/1 Positioning Controller

Limitations: +9 to +24 VDC, RS-232 serial interface 1 kHz Update capability
Max rotation rate of 16000 rpm
Max continuous current of .614 A
Max torque $4.98x10^{-3}$ Nm
Torque Constant of $8.11x10^{-3}$ Nm/A

E. SENSORS

LiDAR Sensor:

Manufacturer: SICK AG, Model # LD-OEM 1000
http://www.sick.com/home/en.html

Limitations: +24 VDC ± 20%, RS-232 serial interface

iGPS:

Manufacturer: Metris
http://www.metris.com/

Limitations: Range between 2 and 40 m
Requires Windows XP based PC running Workspace software

Three-Axis Accelerometer:

Manufacturer: Crossbow, Model # CXL02TG3
http://www.xbow.com/

Limitations: +5 VDC, 2 pin in and 4 pin out (3 axis and temp)
$8.5x10^{-3}$ g bias stability and ±2 g input range

Single Axis Fiber-Optic Rate Gyro:

Manufacturer: KVH Industries, Inc., Model # DSP-3000
http://www.kvh.com/

Limitations: Digital, 100 Hz asynchronous communication via RS-232 interface at 38,400 baud, + 5 VDC, input rate up to ± 375 deg/sec, Offset bias ± 20 deg/hr

F. COMMAND AND DATA HANDLING

108 Mbps Wireless Router:

Manufacturer: Netgear, Model # WGT624 v2
http://www.netgear.com/

Limitations: + 12 VDC
Provides both wired and wired TCP/IP routing

MATLAB XPC Target based Control Computer:

Manufacturer: Diamond Systems, Prometheus Development Kit
http://www.diamondsystems.com/

Limitations: + 5 VDC

Provides near real-time control through embedded MATLAB XPC Target

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. PROCEDURES FOR SETTING UP THE PROMETHEUS CONTROL COMPUTER

The following details the setup procedures for the Prometheus CPU to include installation of the necessary MATLAB XPC Target code for the spacecraft simulators associated with the AMPHIS testbed:

1. Turn off a desktop PC, disconnect the power cord from the power source and open its case.

2. In the current desktop PCs in the SRL, there is only one standard IDE port on the motherboard due to the use of SATA hard drives. For this reason, disconnect the CD-ROM drive from the standard IDE port on the motherboard and then connect a standard IDE cable from the motherboard.

3. Connect other end of the IDE cable to the Prometheus Development Kit ACC-IDEEXT. Connect a small drive power connector from the desktop ATX power supply to the ACC-IDEEXT ensuring the red wire on the small drive connector matches up with the label on the ACC-IDEEXT for +5 V.

4. Install flashdisk in the ACC-IDEEXT in the labeled position.

5. Insert MS-DOS 6.22 boot disk into desktop PC 3 1/2 inch floppy drive.

6. Reconnect the power cord to the desktop PC and boot it.

7. At A:\ prompt, type DIR C: to verify that the flashdisk has been recognized and allocated to the C: drive. This is normal procedure for the desktop PC. If not, find the drive that does correspond to the flashdisk which should be the only one that is 32 MB.

8. At A:\ prompt, type FORMAT C: /S

9. Answer yes to all questions and name volume if desired.

10. Remove the MS-DOS boot disk from the 3 1/2 inch drive and reboot the desktop computer using Ctrl-Alt-Del keys.

11. Start the most recent edition of MATLAB residing on the desktop PC.

12. At the prompt in the MATLAB command window, type xpcexplr and then expand the TargetPC1 icon in the XPC Target Hierarchy on the left of the screen.

13. Under the Communication icon select TCP/IP as the Host target communication under Communication protocol. Under the Target PC TCP/IP configuration, use the following:

a. Target PC IP Address: 192.168.1.3

b. TCP/IP target driver: NE2000

c. TCP/IP target port: 22222

d. TCP/IP target bus: PCI

e. LAN subnet mask address:

f. TCP/IP gateway address:

14. Under the Settings icon, set Target RAM size (MB) to Auto and select 16 MB as the maximum model size. Leave all other boxes unchecked.

15. Under the Appearance icon, check the Enable target scope and select none for the target mouse.

16. Insert a blank 3 1/2 inch floppy disk into the desktop PC and then under the Configuration icon, select DOSLoader and then click Create Bootdisk.

17. After creation of the XPC Target bootdisk, go to My Computer and open the 3 1/2 Floppy icon. Copy all files from this disk to the flashdisk that is still connected through the IDE cable to the motherboard. It should be seen under the Hard Disk Drive section of the My Computer window.

18. Eject the XPC Target floppy, turn off the desktop PC and disconnect the power cord from the power source. Disconnect the IDE cable attached to the ACC-IDEEXT from the motherboard and reconnect the IDE cable between the CD-ROM and the motherboard. Disconnect the small driver power connector from the ACC-IDEEXT.

19. Remove the flashdisk from the ACC-IDEEXT, being careful not to bend the pins on the ACC-IDEEXT.

20. Install the flashdisk in the Prometheus main board. Assemble the Prometheus according the manufacturers directions.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. PROCEDURES FOR INSTALLATION OF IGPS SOFTWARE ON WINDOWS XP BASED COMPUTER

The following details the procedures for installation of the iGPS software on a Windows XP based computer as well as the necessary procedures for uploading new firmware to the iGPS receiver.

1. Prior to installation, ensure that the following files are available on the new computer. These reside in the iGPS folder on the \\special.ern.nps.edu\srl$ network drive on the Naval Postgraduate School.

   a. dotnet.exe

   b. WorkSpace v 6.0.32.1.exe

   c. ATX 1639.asd

   d. TX 2140.asd

   e. Firmware files for current version of WorkSpace

2. Install the Microsoft .NET framework (dotnet.exe) and then restart the computer.

3. Install the iGPS software (WorkSpacev6.0.31.1.exe).

4. Find and run the RegAsmAll.bat file (\Program Files\Arc Second\WorkSpace\RegAsmAll.bat).

5. Start the WorkSpace software (start → All Programs → Arc Second → Workspace). Select unblock if the Microsoft Firewall warning displays.

6. Right Click on the Setup Plans icon under the Objects panel on the left side 3DI Visualization window.

7. Name the Setup Plan Name and click **Next**.

8. Click on the **Load from ASD File…**icon and browse to where the ***.asd** files from step one are located, select the ATX 1639.asd file and click **Open**.

9. Repeat Step 8 but select the TX 2140.asd file and then click **Next**.

10. Select the radio button for “Initially place transmitters in this layout”, ensure **Linear** is selected in the pull down menu and enter 5 meters for the separation (default units is meters) and then click **Next**.

11. Deselect “Automatically create interdetector scale bars for Vector Bar and Setup Bar observations” and “Immediately ‘Run’ this setup plan to begin taking setup observations” and then click **Finish**.

12. Select the 3DI window from the bottom left of the screen and expand the Conductor title in the left panel.

13. Select Configurations and then click **Add** and rename it as desired. Expand the newly created and named configuration, select Transmitters, click on the **Import** icon and browse again to the location of the *.asd files. Select the ATX 1639.asd file and click **Open**. Repeat procedure to import the TX 2140.asd file.

14. Returning to the left panel, select Sensors and then click **Add**. Rename the sensor to an appropriate name (i.e. AMPHIS 1, AMPHIS 2, etc.). Ensure that Sides32 is selected in the Type pull-down menu, leave the remaining boxes deselected and click **OK**.

15. Returning to the left panel, select Receivers and then click **Add**. Select “Black Sun PCE Receiver”, name it as desired and click **OK**. Select the Parameters tab and adjust the Sensitivity and Noise Floor parameters as required. In the SRL lab, the optimal numbers after experimentation are 550 for Sensitivity and 3000 for Noise Floor which is the max for both parameters. These values directly impact the quality of the streaming information between the sensors and the receivers. Best practice for experimentation involved dimming the overhead fluorescent lights and setting the values as noted. If these need to be adjusted after connection (see Step 18), click restart PCE after the changes are made to upload them to the system.

16. Returning to the left panel, select Servers and then click **Add**. This allows the user to select the type of data needed to be collected for each server. To receive position data select “Single Point Server”, rename it to an appropriate name (i.e. ARV 1, ARV 2, etc.) and click **OK**. Under the General tab for this newly created and named

server, ensure all boxes are checked except “Beep on Precision Point”, change the Max Standard Deviation to .0005 which translates to 50 microradians, and select the appropriate sensor from its pull-down menu.

17. Returning to the left panel, reselect Receivers and under the Sensors box in the Setup tab, click add and select the appropriate sensor from the pull-down menu. Connect the senor to the computer via RS232 cable and select the correct COM port from the pull down list and select the “Connected” box.

18. To calibrate the sensor to the transmitters, select **Setup** from the toolbar at the top of the screen and then select Perform Setup from the pull-down menu. In conducting this setup, one must take at least six measurements in different locations around the floor to include at least two that are specifically bounded by a determined distance.

19. In the 3DI Setup Wizard, click **Next,** then **Take Observation Points**, then **New Sensor Observation**. Position the sensor at the desired location and then click **Begin**. Allow the program to collect at least 200 samples from each transmitter and then click **Stop**. If StdDev1 is below 50 microradians click **Next** to move to collect the next observation point data. If the data remains red or no samples are taken then the values in step 16 need to be investigated or there was movement in the sensor/transmitters during the collection.

20. Once at least six observations are completed, click **Finish** and then **Finish** again. Click **Define Scale**, then click **New** and select which observations were constrained and by what length, then click **OK**, and then click **Next**.

21. Finally, click **Calculate Setup** and ensure all observation points are selected to include the constrained points which should be listed in the Scale Bar box. Click **Next** and then click **Calculate** with “Standard” highlighted in the Algorithm box. If the bundle calculation is successful, click **Accept Bundle**, save **Setup Plan** and then **Finish**.

22. Data will now stream from the server. Ensure prior to shutting down to uncheck the box in Step 17.

23. If new firmware needs to be loaded on the receiver due to either corrupt firmware or a new WorkSpace version. The following should be followed:

Close WorkSpace

Ensure receiver is on and attached through RS232 to the PC

Open firmloader software

Browse to where new firmware files are stored

Erase firmware

Upload new firmware

APPENDIX D. PRE-EXPERIMENTATION SET-UP PROCEDURES

A. REFILLING HIGH PRESSURE GAS CYLINDERS

The following procedures outline the process for filling the 6000 PSI air cylinder attached to the MAKO 5404BA air compressor and then the process for filling the air cylinders for the AMPHIS testbed. The procedures for filling the air cylinder on the AUDASS system are as listed in Ref. [16] using the appropriate fill whip. As a point of caution, the air compressor has the potential to overheat and thus may require frequent cycling to allow time for cooling. As a general rule of thumb, if filling the attached cylinder over 3750 PSI, do not run the compressor for more than 10 minutes at a time, allowing it to cool for an additional 20 minutes.

The tanks for each testbed are refilled using the associated fill whips. Each whip is attached to a single adjustable regulator that limits the amount of pressure delivered by the supply air cylinder. Due to both testbed air cylinder pressure limits being lower than the supply air cylinder capability, it is important to properly set the adjustable regulator before being the refilling process on either testbed's air cylinder. As a safety precaution, each air cylinder has an integrated pressure relief valve that limits the pressure that can be delivered to the cylinder. If this pressure relief valve fails, it is possible to over pressurize the vehicle air tank if the adjustable regulator is not set properly.

1. Operating the MAKO 5404BA Air Compressor

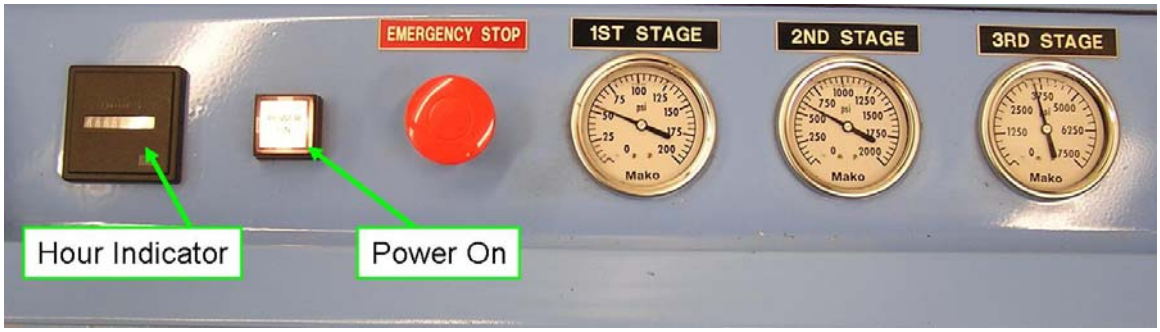

Figure 25. MAKO 5404BA Control Panel

a. Ensure that all breakers inside the left panel of the air compressor are not tripped and the Emergency Stop button is pulled out.

b. Ensure that the supply air cylinder for the SRL is connected to the outlet valve of the air compressor.

c. Don hearing protection and press the Power On button on the main panel of the air compressor. When the air compressor has reached 3750 PSI, begin cycling the compressor on and off as stated above to prevent over-heating of the compressor.

d. The compressor needs to be serviced every 60 hours of use or if the indicator on the filter turns blue.

2. Refilling the AMPHIS Air Cylinders

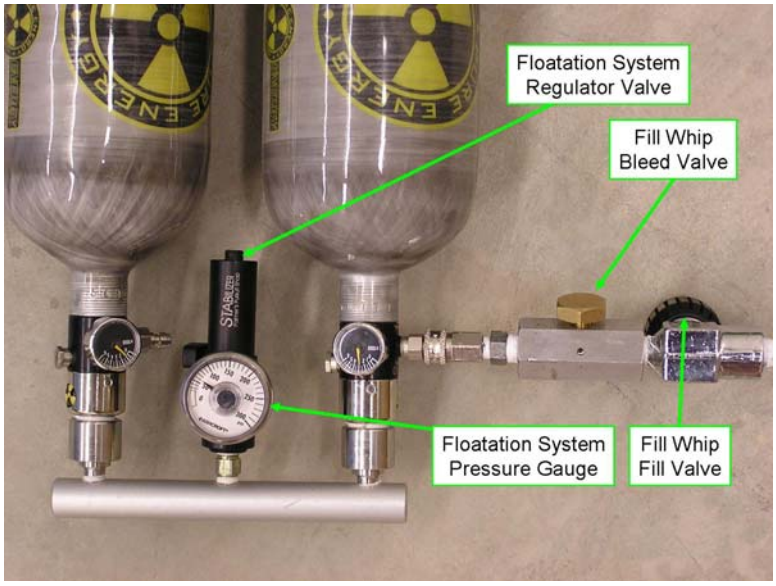

Figure 26. Air Cylinder Refilling Configuration

a. Ensure the supply air cylinder's pressure regulator valve is set to 3000 PSI.

b. Connect the QD style fill whip to one of the air cylinder's fill nozzle. The AMPHIS spacecraft simulator air cylinders do not need to be disconnected from the manifold and regulator assembly.

c. Ensure that both the fill whip bleed valve and the outlet valve from the simulator's regulator are closed.

d. Open the fill valve (turn counter-clockwise) on the whip slowly. If the pressure gauge attached to the outlet of the simulator's floatation system pressure gauge indicates a climbing pressure over 100 PSI, secure the fill valve (full clockwise) and, using the appropriate hex wrench, cycle the regulator's supply pressure valve. During periods of inactivity, the regulator can stick in an open condition and if allowed to remain this way during charging, the pressure gauge could be permanently damaged.

e. After completely filling one air cylinder, close the fill valve (full clockwise) and open the bleed valve (turn counter-clockwise). Disconnect the whip from the air cylinder.

f. Reconnect to the unfilled air cylinder. Proceed through steps 3 through 5 again.

g. Reattach the air cylinders to the simulator.

B. RECHARGING LITHIUM-ION BATTERY PACKS

1. Disconnect battery packs from the simulator

2. Each charge control module has a built-in connector for mating to the lithium-ion battery pack. Properly orient the connector so that pin 1 on the charge cable matches pin 1 on the battery pack and then fully seat the connector into the battery pack.

3. The charge control module will automatically detect the connected battery pack and begin charging. The start of the process can be verified by the presence of two blinking green lights on the charge control module.

4. Upon completion of charging both sections of the battery pack, both green lights on the charge control module will stop flashing and remain solid. Wait until both green lights are solid as the time to complete the charging process for both battery sections is not consistent.

C. VEHICLE START-UP PROCEDURES

Prior to using the AMPHIS spacecraft simulator, ensure that both power cables are properly connected to the battery packs, that the demonstration control computer is

turned on, and that the vehicle is in a safe location with respect to the edges of the simulation floor. Additionally, ensure the master air supply valve is opened and set to 40 PSI.

1. Turn the simulator power switch to the ON position. All components should turn on.

2. After waiting approximately 2 minutes to allow the on-board Versalogic PC-104 to boot and the Netgear wireless router to initialize, connect to the wireless network for the AMPHIS prototype spacecraft simulator (AMPHIS1) from the demonstration computer.

3. Click on the Anyplace Control Admin Module desktop icon on the demonstration computer.

4. When the connection window appears, highlight the computer 192.168.1.2 by clicking on it and the click the connect icon. This will bring up a new screen from which you can control the on-board computer.

APPENDIX E. DEVELOPED MATLAB SCRIPT TO COMMUNICATE WITH MAXON MOTOR AND ENCODER FOR BOTH ROTATING THRUSTER AND MINIATURE-CMG APPLICATIONS

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                         Script to control Micro-CMG                        %
%                           Written by Jason S. Hall                         %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

close all
clear all
clc

%Initialize CMG
s   = openport('COM2',38400);       %Opens port with correct settings
clearfault(s)                       %Clear Any Faults in EPOS 24/1
setenablestate(s)                   %Enables EPOS 24/1
setposreggain(s)                    %Set Pos PID gains (221,270,161)
setvelreggain(s)                    %Set Velocity PI gains  (1267,235)
setcurreggain(s)                    %Set Current PI gains   (465,509)
setaccprofile(s)                    %Set Accel limits (+/-90000)
setmode(s,'pos')
moveposition(s,0,1675*180/pi)       %Set Position at zero

%Import required data for open-loop experiment
gimstate = xlsread('GIMSTATE8');
pos = gimstate(:,1)*180/pi;
vel = gimstate(:,2)*180/pi;

pause(10)                           %Allow time to position simulator
for i = 1:22
    moveposition(s,pos(i),vel(i))   %Moves to position with velocity
end

% Disable CMG
fclose(s)
delete(s)                       %Closes port and deletes port info
clear s


function out = openport(port,baudrate)
%OPENPORT Opens the port attached to the Maxon Motor EPOS 24/1        %
%                                                                     %
%   OUT = OPENPORT('PORT',BAUDRATE) opens the port attached to        %
%to the EPOS 24/1 designated by PORT at the designated BAUDRATE.      %
%Flow Control is set to none, Timeout period is set to 30 seconds     %
%and Request to Send is set to off.                                   %


s               = serial(port);
s.BaudRate      = baudrate;
s.FlowControl   = 'none';
```

```
s.Timeout       = 30;
s.RequestToSend = 'off';

fopen(s)
if s.BytesAvailable
    fread(s,s.BytesAvailable);
end
out = s;


function in = readdata(port,len)
%READDATA Reads in the bytes in the PORT buffer and then outputs the  %
%hexadecimal value of the contents.                                   %
%                                                                     %
%   OUT = READDATA('PORT',len) continues reading the PORT until the   %
%value designated by LEN is reached.  It then takes the value residing%
%in the buffer at this point, converts it to hexadecimal and then     %
%outputs it.  If port is not open already, function will fault out    %

while port.BytesAvailable < len
bytes   = port.BytesAvailable;
end
bytes = port.BytesAvailable;
in = dec2hex(fread(port,bytes));


function out = crccomp(op,dat)
%CRCCOMP Computes the CRC code given an OP and DAT for the EPOS 24/1  %
%                                                                     %
%   OUT = CRCCOMP(op,dat) computes the necessary CRC code and         %
%repackages it so that it in is the proper format for sending out     %
%(i.e. LSB then MSB)                                                  %

m = 2;
d = cellstr(dat);
l = hex2dec(d(1)) + 1;
d(1) = strcat(op,d(1));
for i = 2:l+1
    d(i) = strcat(d(m+1),d(m));
    m = m+2;
end
d(l+2) = cellstr('0000');
d = hex2dec(d(1:l+2));
[m,n]=size(d);
crc     = uint16(0);

for i = 1:m
    shifter = uint16(hex2dec('8000'));
    c   = uint16(d(i));
    while all(shifter)
        carry = bitand(crc,uint16(hex2dec('8000')));
        crc = bitshift(crc,1);
        if all(bitand(c,shifter)), crc = crc + 1; end
        if all(carry), crc = bitxor(crc,uint16(hex2dec('1021'))); end
        shifter = bitshift(shifter,-1);
    end
```

```
end
crc = dec2hex(crc);
if size(crc,2) == 3
    crc = cat(2,'0',crc);
elseif size(crc,2) == 2
    crc = cat(2,'00',crc);
end
crcn(1) = cellstr(crc(3:4));
crcn(2) = cellstr(crc(1:2));
out = crcn;


function out = comm(port,op,dat,len)
%COMM provides framework for sending and receiving information with   %
%   the Maxon Motor EPOS 24/1 using the established protocol          %
%                                                                     %
%   OUT = COMM('PORT',op,dat,len) sends the OP first and receives     %
%acknowledgement then computes the CRC given the OP and DAT using     %
%CRCCOMP over the port, PORT.  If port is not open already, function  %
%will fault out.  It then sends this to the EPOS 24/1 and receives    %
%confirmation.  If the EPOS 24/1 received the command, it returns the %
%value in the buffer.  A correct number of bytes to be in the finished%
%buffer is designated by LEN.                                         %
%                                                                     %
%   Requires CRCCOMP and READDATA residing in directory               %


fwrite(port,hex2dec(op))
in = readdata(port,1);
if in == '4F'
    crc = crccomp(op,dat);
    dat = cat(2,dat,crc);
    fwrite(port,hex2dec(dat))
    in = readdata(port,2);
else if in == '46'
        fprintf('\nInvalid command sent to CMG\n')
        clearfault(port)  %run initialization command again
    end
end

if in(1,:) == '4F'
    if in(2,:) == '00'
        fwrite(port,79)
        in = readdata(port,len);
    else in = fprintf('\nCMG not ready to send\n');
    end
else in = fprintf('\nCMG did not receive command\n');
end
out = in;


function out = clearfault(port)
%CLEARFAULT Clears any faults associated with Maxon Motor EPOS 24/1   %
%                                                                     %
%   OUT = CLEARFAULT('PORT') clears faults of EPOS 24/1 attached to   %
%to port, PORT.  If port is not open already, function will fault out %
```

```
%                                                                       %
%   Requires the COMM, READDATA, and CRCCOMP residing in directory      %

in = comm(port,'10',{'01','41','60','00','01'},11);
if ~all(in == (char(['03';'00';'00';'00';'00';'08';'07';'F3';'01';...
        '85';'39'])))
    fwrite(port,79)
    fprintf('\nEncoder is in a fault state \n')
    in =
comm(port,'11',{'03','40','60','00','01','80','00','00','00'},7);
    if all(in == char(['01';'00';'00';'00';'00';'51';'AA']))
        fprintf('\nEncoder cleared any faults \n')
        fwrite(port,79)
    end
else
    fprintf('\nEncoder is not in fault state \n')
    fwrite(port,79)
end


function out = setenablestate(port)
%SETENABLESTATE Enables the Maxon Motor EPOS 24/1                       %
%                                                                       %
%   OUT = SETENABLESTATE('PORT') enables the EPOS 24/1 attached to      %
%to port, PORT.  If port is not open already, function will fault out   %
%                                                                       %
%   Requires the COMM, READDATA, and CRCCOMP residing in directory      %

in = comm(port,'10',{'01','41','60','00','01'},11);
if in(1:6,:) == ['03';'00';'00';'00';'00';'37']
    fwrite(port,79)
    fprintf('\nEncoder is enabled \n')
else
    fwrite(port,79)
    fprintf('\nEncoder is not enabled \n')
    comm(port,'11',{'03','40','60','00','01','06','00','00','00'},7);
    fwrite(port,79)
    comm(port,'10',{'01','41','60','00','01'},11);
    fwrite(port,79)
    comm(port,'11',{'03','40','60','00','01','0F','00','00','00'},7);
    fwrite(port,79)
    comm(port,'10',{'01','41','60','00','01'},11);
    fwrite(port,79)
    comm(port,'11',{'03','40','60','00','01','0F','01','00','00'},7);
    fwrite(port,79)
    setenablestate(port)
end


function out = setposreggain(port)
%SETPOSREGGAIN Sets the position regulator PID gains on the Maxon       %
%   Motor EPOS 24/1 to experimentally defined values of 221, 270        %
%   and 161 respectively                                                %
%                                                                       %
%   OUT = SETPOSREGAIN('PORT') sets the position gains of the EPOS      %
%   24/1 attached to port, PORT.  If port is not open already, the      %
%   function will fault out                                             %
```

```
%                                                                       %
%   Requires the COMM, READDATA, and CRCCOMP residing in directory      %

comm(port,'11',{'03','FB','60','01','01','DD','00','00','00'},7);
fwrite(port,79)
comm(port,'11',{'03','FB','60','02','01','0E','01','00','00'},7);
fwrite(port,79)
comm(port,'11',{'03','FB','60','03','01','A1','00','00','00'},7);
fwrite(port,79)
fprintf('\nPosition Regulator Proportional Gain set to 221, ')
fprintf('Integral Gain to 270 and Derivative Gain to 161 \n');


function out = setvelreggain(port)
%SETVELREGGAIN Sets the velocity regulator PI gains on the Maxon        %
%   Motor EPOS 24/1 to experimentally defined values of 1267 and        %
%   235 respectively                                                    %
%                                                                       %
%   OUT = SETVELREGAIN('PORT') sets the velocity gains of the EPOS      %
%   24/1 attached to port, PORT.  If port is not open already, the      %
%   function will fault out                                             %
%                                                                       %
%   Requires the COMM, READDATA, and CRCCOMP residing in directory      %

comm(port,'11',{'03','F9','60','01','01','F3','04','00','00'},7);
fwrite(port,79)
comm(port,'11',{'03','F9','60','02','01','EB','00','00','00'},7);
fwrite(port,79)
fprintf('\nVelocity Regulator Proportional Gain set to 1267, ')
fprintf('Integral Gain to 235 \n');


function out = setcurreggain(port)
%SETCURREGGAIN Sets the current regulator PI gains on the Maxon         %
%   Motor EPOS 24/1 to experimentally defined values of 465 and         %
%   509 respectively                                                    %
%                                                                       %
%   OUT = SETCURREGAIN('PORT') sets the current gains of the EPOS       %
%   24/1 attached to port, PORT.  If port is not open already, the      %
%   function will fault out                                             %
%                                                                       %
%   Requires the COMM, READDATA, and CRCCOMP residing in directory      %

comm(port,'11',{'03','F6','60','01','01','D1','01','00','00'},7);
fwrite(port,79)
comm(port,'11',{'03','F6','60','02','01','FD','01','00','00'},7);
fwrite(port,79)
fprintf('\nCurrent Regulator Proportional Gain set to 465, 509')


function out = setaccprofile(port)
%SETACCPROFILE Sets the Acceleration and Decelleration limits for       %
%   the Maxon Motor EPOS 24/1 to +/- 90,000 rpm^2                       %
%                                                                       %
%   OUT = SETACCPROFILE('PORT') enables the EPOS 24/1 attached to       %
%to port, PORT.  If port is not open already, function will fault out   %
%                                                                       %
```

```
%   Requires the COMM, READDATA, and CRCCOMP residing in directory    %

comm(port,'11',{'03','83','60','00','01','90','5F','01','00'},7);
fwrite(port,79)
comm(port,'11',{'03','84','60','00','01','90','5F','01','00'},7);
fwrite(port,79)
fprintf('\nProfile Velocity set to 16000 rpm, ')
fprintf('Acceleration and Decceleration set to 90,000 rpm/s\n')


function out = setmode(port,type)
%SETMODE Sets the mode on the Maxon Motor EPOS 24/1                    %
%                                                                      %
%   OUT = SETMODE('PORT',TYPE) sets the of mode of the EPOS 24,1       %
%attached to port, PORT.  If port is not open already, the function    %
%will fault out.  TYPE is a decimal number that corresponds to the     %
%following modes:                                                      %
%                                                                      %
%   1   -   Position Profile Mode                                      %
%   3   -   Velocity Profile Mode                                      %
%                                                                      %
%   Requires the COMM, READDATA, and CRCCOMP residing in directory     %

if type == 'vel'
    type = 3;
elseif type == 'pos'
    type = 1;
end

type = dec2hex(type);
dat =
cat(2,{'03','60','60','00','01'},cellstr(cat(2,'0',type)),'00','00','00
');
comm(port,'11',dat,7);
fwrite(port,79)


function out = moveposition(port,pos1,vel1)
%MOVEPOSITION Rotates the Gimbal to the defined position at the        %
%   defined speed via the Maxon Motor EPOS 24/1                        %
%                                                                      %
%   OUT = MOVEPOSITION('PORT',POS1,VEL1) rotates the gimbal motor      %
%via the EPOS 24,1 attached to port, PORT.  If port is not open        %
%already, the function will fault out.  POS1 is a decimal number       %
%that represents the number of absolute to rotate to.  VEL1 is a       %
%decimal number that corresponds to speed in degrees/s.                %
%                                                                      %
%   Requires the COMM, READDATA, and CRCCOMP residing in directory     %


pos = floor(pos1*pi/180*(44000/(pi/2)));
poscheck = 0;
if pos1 < 0
    pos = 65535 - abs(pos);
    if pos < 0
    poscheck = 1;
```

```
    pos = 65535 - abs(pos);
    end
end
mp      = dec2hex(abs(pos));
if size(mp,2) == 2
    mpn(1) = cellstr(mp);
    mpn(2) = cellstr('00');
elseif size(mp,2) == 1
    mpn(1) = cellstr(cat(2,'0',mp));
    mpn(2) = cellstr('00');
elseif size(mp,2) == 3
    mpn(1) = cellstr(mp(2:3));
    mpn(2) = cellstr(cat(2,'0',mp(1)));
else
    mpn(1) = cellstr(mp(3:4));
    mpn(2) = cellstr(mp(1:2));
end
mpn;

vel =   floor(vel1*pi/180*16000/6.95);
vp  =   dec2hex(abs(vel));
if size(vp,2) == 2
    vpn(1) = cellstr(vp);
    vpn(2) = cellstr('00');
elseif size(vp,2) == 1
    vpn(1) = cellstr(cat(2,'0',vp));
    vpn(2) = cellstr('00');
elseif size(vp,2) == 3
    vpn(1)  = cellstr(vp(2:3));
    vpn(2) = cellstr(cat(2,'0',vp(1)));
else vpn(1)  = cellstr(vp(3:4));
    vpn(2) = cellstr(vp(1:2));
end
vpn;

dat = cat(2,{'03','81','60','00','01'},vpn,{'00','00'});
comm(port,'11',dat,7);
fwrite(port,79)

if pos1 >= 0
   dat = cat(2,{'03','7A','60','00','01'},mpn,{'00','00'});
elseif poscheck
    dat = cat(2,{'03','7A','60','00','01'},mpn,{'FE','FF'});
else
    dat = cat(2,{'03','7A','60','00','01'},mpn,{'FF','FF'});
end
comm(port,'11',dat,7);
fwrite(port,79)
comm(port,'11',{'03','40','60','00','01','3F','00','00','00'},7);
fwrite(port,79)
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX F. MATLAB CODE AND SIMULINK MODEL USED FOR AMPHIS PROTOTYPE SPACECRAFT SIMULATOR SLEW MANUEVER SIMULATION

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%           Script to Simulate Slew Manuever for AMPHIS Simulator     %
%                          Written by Jason S. Hall                   %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all
clc
close all

global CONST VAR

%Unit Conversion
in2m    = .0254;
lb2kg   = .4536;

%Vehicle Parameters (English units)
CONST.m = 52.8;                     %lb
CONST.x = 12;                       %inches
CONST.y = 12;                       %inches
CONST.z = 27.25;                    %inches

%Vehicle Parameters (SI units)
CONST.x = CONST.x*in2m;             %meter
CONST.y = CONST.y*in2m;             %meter
CONST.z = CONST.z*in2m;             %meter
CONST.m = CONST.m*lb2kg;            %kg

%Moment of Inertia of Vehicle Calculation
Jx      = 1/3*CONST.m*(CONST.y^2+CONST.z^2);
Jy      = 1/3*CONST.m*(CONST.x^2+CONST.z^2);
Jz      = 1/3*CONST.m*(CONST.x^2+CONST.y^2);
%Jz      = .25;

CONST.J = diag([Jx;Jy;Jz]);         %kg-m^2

%CMG Parameters (SI units)
CONST.mw  = .112;                   %kg
CONST.dw  = .053;                   %m
CONST.Jw  = 1/2*CONST.mw*(CONST.dw/2)^2; %kg-m^2
CONST.OMw = 12000*2*pi/60;          %rad/s
CONST.h_w = CONST.Jw*CONST.OMw;     %N-m-s
CONST.gimsat = 89*pi/180;           %rad
CONST.gimratesat = 16000*2*pi/60;   %rad/s
CONST.Jg  = 37.17e-6;               %kg-m^2
CONST.Tgimmax = 4.055e-3;           %N-m
CONST.Tmaxz=CONST.h_w*CONST.gimratesat;   %N-m

%Initial Conditions
```

```
PSIo  = 0;
omzo  = 0;
delo  = 0;
deldo = 0;
VAR.Xo =  [PSIo;omzo;delo;deldo];

%Final Conditions
s = 30;
PSIf = 1*pi/180*s;
omzf = 0;
VAR.Xref = [PSIf;omzf];

%Calculation of Gains for PD and Quaternion_Feedback Control Law
zeta = .9;

if PSIf < 15*pi/180
    Kpz = CONST.Tmaxz/(10*pi/180)/5000;
    om_nz = sqrt(Kpz/Jz);
    Kdz=2*Jz*om_nz*zeta;
    Kpgim = CONST.Tgimmax/(10*pi/180)/300;
    omgim = sqrt(Kpgim/CONST.Jg);
    Kdgim = 2*CONST.Jg*omgim*zeta;
elseif PSIf < 30*pi/180
    Kpz = CONST.Tmaxz/(10*pi/180)/5000;
    om_nz = sqrt(Kpz/Jz);
    Kdz=2*Jz*om_nz*zeta;
    Kpgim = CONST.Tgimmax/(10*pi/180)/300;
    omgim = sqrt(Kpgim/CONST.Jg);
    Kdgim = 2*CONST.Jg*omgim*zeta;
elseif PSIf < 60*pi/180
    Kpz = CONST.Tmaxz/(10*pi/180)/6000;
    om_nz = sqrt(Kpz/Jz);
    Kdz=2*Jz*om_nz*zeta;
    Kpgim = CONST.Tgimmax/(10*pi/180)/150;
    omgim = sqrt(Kpgim/CONST.Jg);
    Kdgim = 2*CONST.Jg*omgim*zeta;

elseif PSIf < 90*pi/180
    Kpz = CONST.Tmaxz/(10*pi/180)/5000
    om_nz = sqrt(Kpz/Jz);
    Kdz=2*Jz*om_nz*zeta;
    Kpgim = CONST.Tgimmax/(10*pi/180)/300;
    omgim = sqrt(Kpgim/CONST.Jg);
    Kdgim = 2*CONST.Jg*omgim*zeta;
else Kpz = CONST.Tmaxz/(10*pi/180)/5000
    om_nz = sqrt(Kpz/Jz);
    Kdz=2*Jz*om_nz*zeta;
    Kpgim = CONST.Tgimmax/(10*pi/180)/300;
    omgim = sqrt(Kpgim/CONST.Jg);
    Kdgim = 2*CONST.Jg*omgim*zeta;
end

VAR.Ksc = [Kpz;Kdz];
VAR.Kgim = [Kpgim;Kdgim];

sim('test7',60)
```

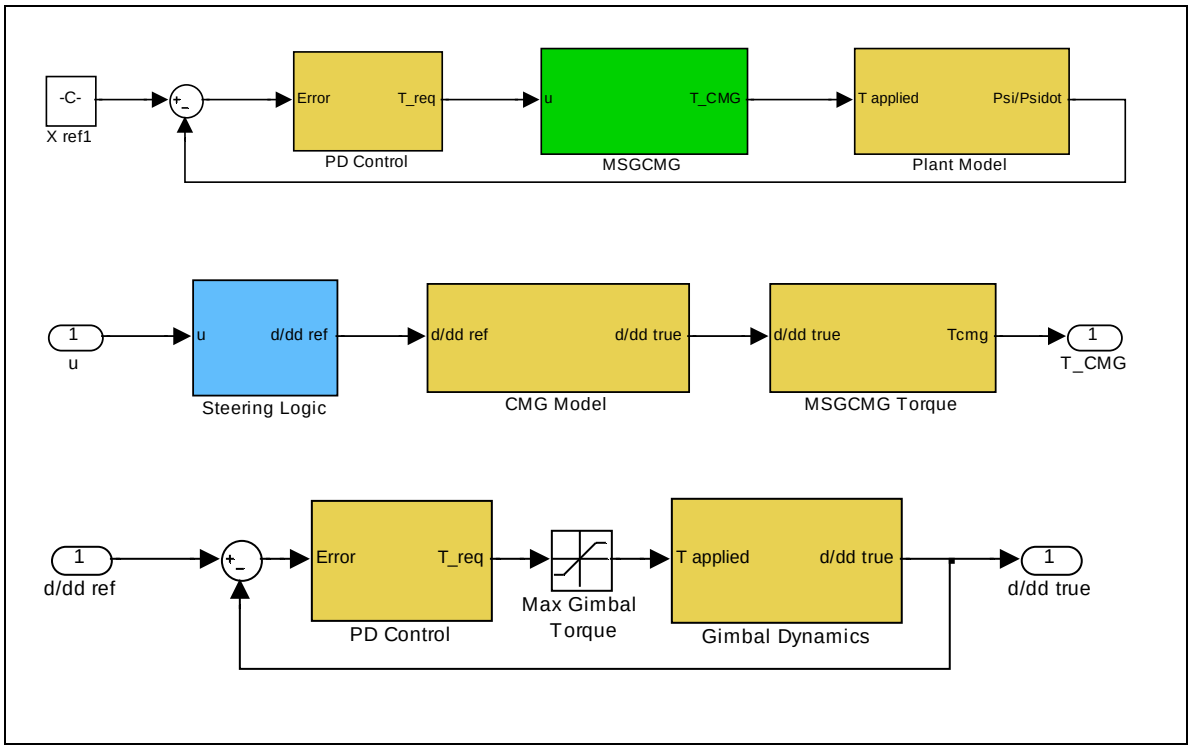

Figure 27. SIMULINK Block Diagrams to Simulate a Slew Manuever

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

1. Hearings before the House Armed Services Committee on Space Acquisition, 109th Cong. 2d Sess. 3(testimony of Dr. Pedro "Pete" L. Rustan, Director, Advanced Systems and Technology at the National Reconnaissance Office).

2. Mathieu, C., & Weigel, A.L. (2005). Assessing the Flexibility Provided by Fractionated Spacecraft. *AIAA Space 2005 Conference*.

3. Dubowsky, Steven. *The Planning and Control of Space Robotic Systems*. Retrieved August 23, 2006, from http://robots.mit.edu/projects/jaxa/index.html.

4. Lichter, M.D., Dubowsky, S., Ueno, H., & Mitani, S. (2005). Shape, Motion, and Parameter Estimation of Flexible Space Structures using Laser Rangefinders. *Proc. Robotics: Science and Systems I*. Cambridge, MA.

5. Ueno, H. & Oda, M. (2004) Assembly Robot Ground Experiments for Space Solar Power System in JAXA. *Earth And Space 2004 Conference.*

6. Ueno, H. & Oda, M. (2004). Strategy and Ground Experiments on Space Flexible Structure Tracking by Multiple Robots on Vibrating Bases. *Proc. Romansy Conference*. Montreal, Canada.

7. Regehr, M., Acihese, A., Ahmed, A., Aung, M., Bailey, R., Bushnell, C., Clark, K., Hicke, A., Lytlel, B., MacNeal, P., Rasmussen, R., Shields, J., & Singh, G. (2004). The Formation Control Testbed. *Proc 2004 IEEE Aerospace Conference*. Big Sky, MT.

8. Lappas, V.J., Steyn, W.H., & Underwood, C.I. (2002) Practical Results on the Development of a Control Moment Gyro based Attitude Control System for Agile Small Satellites. *AIAA/USU Small Satellite Conference*. Logan, UT.

9. Roser, X., & Sghedoni, M. (1997). Control Moment Gyroscopes (CMG's) and their application in future scientific missions. *Proc. 3rd International Conf. on Spacecraft Guidance, Navigation and Control Systems*. pp. 523-598.

10. Lappas, V.J. (2006) Attitude Control for Planetary Missions. *Presentation for Attitude Control Technology Workshop*. Surrey Space Centre.

11. Leloglu, U., & Sweeting, M. (2002) BILSAT-1: A Case Study For The Surrey Satellite Technology Ltd Know-How Transfer And Training Programme. *53rd International Astronautical Congress*. Houston, TX.

12. Peck, M. (2005) Low-Power, High-Agility Space Robotics. *AIAA Guidance, and Control Conference and Exhibit*. San Francisco, CA.

13. Romano, M., Friedman, D., & Shay, Tracy. (2005) Laboratory Experimentation of Autonomous Spacecraft Approach and Docking to a Collaborative Target. Accepted for publication. To Appear AIAA Journal of Spacecraft and Rockets.

14. Porter, M. (2002). Development and Control of the Naval Postgraduate School Planar Autonomous Docking Simulator (NPADS). MS Thesis, Dept. of Astro. and Mech. Eng., NPS.

15. Friedman, D. (2005). Laboratory Experimentation of Autonomous Spacecraft Docking Using Cooperative Vision Navigation. MS Thesis, Dept. of Astro. And Mech. Eng., NPS.

16. Shay, T.J. (2005). Design and Fabrication of Planar Autonomous Spacecraft Simulator with Docking and Fluid Transfer Capability. MS Thesis, Dept. of Astro. and Mech. Eng., NPS.

17. Price, Bill. (to be completed December 2006). MS Thesis, Dept. of Astro and Mech Eng., NPS.

18. Wie, B. (1998). Space Vehicle Dynamics and Control. *AIAA Educational Series*. Tempe, AZ.

19. Maxon Motor (2004). *EPOS Positioning Controller Communication Guide.*

20. SICK. (2004). *LD Laser Scanners Product Information.*

21. Eikenberry, Blake. (to be completed December 2006). MS Thesis, Dept. of Astro and Mech Eng., NPS.

22. Inalhan, G., Busse, F., & How, J. (2000) Precise Formation Flying Control of Multiple Spacecraft Using Carrier-Phase Differential GPS. *Proc. AAS/AIAA Space Flight Mechanics Meeting (AAS 00-109).* Clearwater, FL.

23. Arc Second. (2005). *Indoor GPS Workspace, Advanced Metrology Edition, User's Guide Version 6.0.*

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Marcello Romano
   Naval Postgraduate School
   Monterey, CA